

Р.А. Кадиркулов, А.Д. Рыскулбекова,
Г.К. Нурмуханбетова

ИНФОРМАТИКА

Учебник для учащихся 8 класса
общеобразовательной школы

Рекомендовано
Министерством образования и науки
Республики Казахстан

АЛМАТЫКІТАП БАСПАСЫ
2021

УДК 373.167.1
ББК 32.973 я 72
К 79

Условные обозначения

	– цель урока		– индивидуальная работа
	– подумай		– вопросы для размышления
	– новые знания		– задание
	– практическая работа		– домашнее задание
	– анализ		– словарь
	– синтез		– задание легкой сложности
	– оценка		– задание средней сложности
	– работа в группе		– задание высокой сложности
	– работа в паре		– задание на CD

В учебно-методический комплекс «Информатика» входят учебник и методическое руководство для учителя. Электронный учебник размещен на платформе издательства www.topiq.kz. Чтобы работать независимо от интернета, к учебнику прилагается CD.

Кадиркулов Р.А. и др.

К 79 Информатика: Учебник для учащихся 8 класса общеобразовательной школы / Р.А. Кадиркулов, А.Д. Рыскулбекова, Г.К. Нурмуханбетова. – Алматы: АЛМАТЫКІТАП БАСПАСЫ, 2021. – 152 с.: илл.

ISBN 978-601-01-4418-7

УДК 373.167.1
ББК 32.973 я 72

ISBN 978-601-01-4418-7

© Кадиркулов Р.А., Рыскулбекова А.Д.,
Нурмуханбетова Г.К., текст, 2021
© ТОО «АЛМАТЫКІТАП БАСПАСЫ», 2021

Дорогие ученики!

В этом учебном году вы изучите основы компьютерных устройств и сетей, алфавитные методы измерения информации, работу с электронными таблицами, научитесь создавать алгоритмы и программы на языке программирования. Приобретая новые знания и опыт, вы будете находить ответы на различные вопросы, делать выводы в ходе совместных обсуждений и выполнять практические и творческие задания. Вы узнаете, как анализировать материал, обобщать его, будете представлять свои проекты, доказывать их практическое применение. Интересными и полезными для вас будут изучение алгоритмов, моделирования, дизайн, исследовательские задачи и задания для развития критического и логического мышления.

Новые темы в учебнике изложены доступно: от простого к сложному. Большинство из них направлено на программирование и алгоритмизацию. Роль алгоритмов в современном обществе определяется не только техническими аспектами их использования. Алгоритмический подход является неотъемлемой частью повседневной жизни и работы. Во многих случаях результат деятельности зависит от того, насколько точно человек знает алгоритмический смысл своих действий: Что делать в данное время? В каком порядке? Каким должен быть результат действий?

В учебнике предложены методы разработки алгоритма, концепции написания и рассмотрены особенности самого популярного языка программирования Python. Надеемся, что знания и новый опыт, приобретенные в области компьютерных наук в цифровую эпоху, пригодятся вам в будущем.

Постарайтесь всесторонне изучить предмет информатики. Помните, что знания, которые вы получаете, требуют энтузиазма, внимания и настойчивости.

Желаем удачи в учебе!

Раздел I Технические характеристики компьютера и сетей

1.1 Измерение информации. Алфавитный подход



Как применить алфавитный подход к измерению объема информации?



Подумай

- Что такое алфавитный подход к измерению объема информации?
- В каких случаях он используется?



Новые знания

Алфавитный подход

Алфавитный подход к измерению информации эквивалентен вероятностному подходу. Здесь рассматривается измерение количества информации текстового (символьного) сообщения, составленного из символов какого-нибудь алфавита. При алфавитном подходе к измерению информации количество информации не зависит от содержания. Алфавитный подход – единственный способ, дающий возможность измерения информации, обрабатываемой на компьютерах, вычислительной технике, мобильных устройствах.

А	Б	В	Г	Д
Е	Ё	Ж	З	И
Й	К	Л	М	Н
О	П	Р	С	Т
У	Ф	Х	Ц	
Ч	Ш	Щ	Ъ	
Ы	Ь	Э	Ю	Я

Основным понятием данного подхода является алфавит. **Алфавит** – ограниченное множество символов, предназначенных для представления информации. Число символов алфавита называют **мощностью алфавита**.



Если считать, что алфавит – это событие, то каждый из символов в сообщении можно рассматривать как одно из состояний события. Если появление символов равновероятно, то можно рассчитать, сколько бит информации несет каждый символ (X) с помощью уравнения Хартли.

$$N = 2^X$$

N – мощность используемого алфавита;
 X – информационный вес символа.

Отсюда можно сделать вывод: для вычисления информационного объема сообщения I , состоящего из K символов, необходимо информационный вес одного символа X умножить на K .



информационный вес – ақпараттық салмақ – information weight
алфавитный метод – алфавиттік әдіс – alphabetic method

$$I = K \cdot X$$

I – информационный объем сообщения;
 K – количество символов в сообщении;
 X – информационный вес символа.

Величину количества информации можно назвать **информационным объемом** текста. Наименьшая мощность алфавита для передачи информации равна 2. Такой алфавит называется **двоичным**.

В двоичном алфавите информационный вес символа равен одному биту, так как $2^x = 2$. Таким образом, один символ двоичного алфавита несет 1 бит информации. В дальнейшем более подробно рассмотрим алфавит внутреннего (машинного) языка компьютера – язык двоичного кодирования.

Бит – основная единица измерения информации. **Байт** – информационный вес символа алфавита. В современных вычислительных системах байт состоит из восьми битов и, соответственно, может принимать одно из 256 (2^8) различных значений (состояний, кодов).

Количество информации зависит не от содержания текста, а от его объема (от числа знаков в тексте) и от мощности алфавита.

Пример. Сообщение, состоящее из 20 символов, записано буквами алфавита, представленного 64 символами. Какое количество информации оно несет? Ответ: 120 бит информации.

Примечание. Здесь задан алфавит мощностью $N = 64$, количество символов $K = 20$. Тогда по формуле Хартли информационный вес (X) составляет 6 бит, количество информации (I) составляет: $20 \cdot 6 = 120$ бит.



Анализ



Проанализируйте образцы заданий и их способы решения. Поделитесь своими мыслями.

Пример № 1

Сообщение, содержащее 40 символов, составлено с помощью 32-символьного алфавита. Каков информационный объем сообщения?

Дано:

$$N = 32$$

$$K = 40$$

Найти: I – ?

Решение:

$$N = 2^x \Rightarrow 32 = 2^x \Rightarrow X = 5;$$

$$I = K \cdot X \Rightarrow I = 40 \cdot 5 \Rightarrow I = 200 \text{ бит} = 40 \text{ байт}$$

Примечание: Разделили информационный объем 200-битного сообщения на вес 5-битной информации и выразили его в байтах.

Ответ: 40 байт.

Пример №2

Электронная книга содержит 150 страниц, на каждой странице 40 строк, в каждой строке 60 символов. Определите информационный объем книги.

Дано:

$$\begin{aligned} K_1 &= 150 \\ K_2 &= 40 \\ K_3 &= 60 \\ N &= 256 \end{aligned}$$

Найти: $I - ?$ **Решение:**

Формула для вычисления информационного объема: $I = K \cdot X$

$$N = 2^x = > 256 = 2^x = > X = 8 \text{ бит};$$

$$K = K_1 \cdot K_2 \cdot K_3; K = 150 \cdot 40 \cdot 60; K = 360\,000;$$

$$I = 8 \text{ бит} \cdot 360\,000 = 2\,880\,000 \text{ бит} = 360\,000 \text{ байт} = 351,56 \text{ Кб.}$$

Примечание. Мощность алфавита, используемого в компьютере, равна 256.

Ответ: 351,56 Кб**Пример №3**

Алфавит казахского языка из 42 букв и добавленные к ним 12 знаков препинания и 10 цифр представляют собой 64-символьный алфавит. Алфавит английского языка состоит из 32 символов, включающих 26 букв и 6 знаков препинания. Сравните объем информации в сообщениях, составленных из 70 символов алфавита казахского языка и из 70 символов алфавита английского языка.

Дано:

$$\begin{aligned} N_1 &= 64 \\ N_2 &= 32 \\ K_1 &= 70 \\ K_2 &= 70 \end{aligned}$$

Найти: $\frac{I_1}{I_2} - ?$ **Решение:**

Формула для вычисления информационного объема: $I = K \cdot X$

$$N_1 = 2^{X_1} = > 64 = 2^{X_1} = > X_1 = 6 \text{ бит};$$

$$N_2 = 2^{X_2} = > 32 = 2^{X_2} = > X_2 = 5 \text{ бит};$$

$$I_1 = 70 \cdot 6 \text{ бит} = 420 \text{ бит}; \quad I_2 = 70 \cdot 5 \text{ бит} = 350 \text{ бит}$$

$$\frac{I_1}{I_2} = \frac{420}{350} = 1,2$$

Ответ: больше в 1,2 раза для казахского алфавита.

**Синтез**

Составьте текст задачи на измерение количества информации алфавитным подходом. При составлении текста сделайте акцент на актуальность их содержания.



Оценка



Какое значение имеет алфавитный подход измерения информации?



Вопросы для размышления

1. Что такое алфавит?
2. Что определяет мощность алфавита?
3. Что такое информационный вес символа алфавита?
4. Как можно измерить информационный объем текста с помощью алфавитного подхода?
5. Какие формулы и величины используются при алфавитном подходе измерения информации?
6. Что такое байт, килобайт, мегабайт, гигабайт?



Задание

1. В Южной Америке на берегах реки Амазонки найдено неизвестное до сих пор племя индейцев. Алфавит племени состоит только из 8 букв. Какой объем информации несет одна буква алфавита?
2. История развития планеты Земля показывает, что на ней существовали и исчезли несколько цивилизаций. Согласно дошедшей до нас информации, в одной из этих цивилизаций информационный объем одного символа составлял 6 бит. Определите количество символов алфавита потерянной цивилизации.
3. Индейское племя, имеющее алфавит из 128 символов, отправило письмо, состоящее из 30 букв, с гонцом в соседнее племя. Какой объем информации доставил гонец?
4. Определите количество килобайт в сообщении из 512 символов, составленном с помощью 16-символьного алфавита.



Домашнее задание

Тема исследования	Направления и вопросы для исследования
Подходы к измерению информации	<ul style="list-style-type: none"> • История возникновения измерения информации с древности до наших дней. • Информация, которая нас окружает. • Единицы измерения информации. • Методы измерения информации.

Раздел I Технические характеристики компьютера и сетей

1.2 Измерение информации. Вероятностный подход



Как применить вероятностный подход к измерению объема информации?



Подумай

- Для чего нужно определять объем информации?
- Какие виды измерения объема информации вы знаете?



Новые знания

Измерение информации

Объем информации в сообщении о каких-либо событиях зависит от вероятности его получения. Имеются различные устройства и системы, которые дают возможность собрать и обработать информацию.

Сообщение – это информация, представленная в определенной форме, которая в процессе передачи данных поступает к получателю.

Сообщение – это слова, которые мы слышим (радиопередачи, объяснение учителя), видеоизображение, которое мы визуальное воспринимаем (фильм, сигнал светофора), текст книги, который мы читаем, и т.д.

Информативность сообщения характеризуется содержащейся в нем полезной информацией, для определения объема которой необходима единица измерения. Получение новой информации приводит к расширению знаний человека или к уменьшению неопределенности. Есть два метода измерения информации: алфавитный и вероятностный (рис. 1).



Рис. 1. Методы измерения информации



вероятностный метод – ықтималдық әдісі – probabilistic method
 мощность алфавита – алфавиттің қуаттылығы – power of the alphabet

Вероятностный подход определения объема информации

Сообщение, уменьшающее неопределенность знания в два раза, несет **1 бит** информации.

Сообщение об одном из двух равновероятных результатов некоторого события несет **1 бит** (условно 0 или 1) информации.

Пример 1. Подбрасывая монету, вы загадываете, что выпадет: орел или решка. Есть всего два возможных результата. Ни один из этих результатов не имеет преимущества перед другим. В таком случае говорят, что они равновероятны. Узнав результат подбрасывания монеты, вы получили один бит информации.



Примечание. Определение «бит» в данной формулировке может быть сложным для восприятия, потому что содержит непонятный термин «неопределенность знаний». Рассмотрим примеры, чтобы раскрыть его смысл.

Так как у монеты есть две стороны и мы не знаем, какой стороной она упадет, то перед ее подбрасыванием **неопределенность знаний** равна двум. Следовательно, результатом будет решение уравнения $2^x = 2$. Получаем, что $X = 1$ бит.

Пример 2. В игре в кости используют игральный кубик с шестью гранями. Сколько бит информации несет игроку каждый бросок кубика?

Примечание. Каждая из шести сторон кубика имеет **равную вероятность** выпадения. Тогда неопределенность знаний о результате броска кубика уменьшится в 6 раз. Количество информации определится при решении уравнения $2^x = 6$.



Вероятностное событие с ограниченной возможностью числа N исходов считается *случайным событием*.

В 1928 г. американский инженер **Ральф Хартли** рассматривал процесс получения информации как выбор одного сообщения из конечного заранее заданного множества из N равновероятных сообщений. Пусть в некотором сообщении содержатся сведения о том, что произошло одно из N возможных равновероятных событий. Тогда количество информации, заключенное в этом сообщении, – X бит и число N – количество событий связаны формулой Хартли. С основными особенностями формулы Ральфа Хартли вы познакомитесь в старших классах. В таблице 1 дано количество событий в зависимости от количества бит.

$$N = 2^x$$

N – количество равновероятных событий;
 X – количество бит в сообщении.

Таблица 1. Количество событий в зависимости от количества бит

Количество бит	0	1	2	3	4	5	6	7	8	9	10
Количество событий	1	2	4	8	16	32	64	128	256	512	1024

Примечание. По таблице можно определить количество бит, соответствующее числу событий. Например, на уроке по пятибальной системе можно получить одну из четырех оценок: «5», «4», «3», «2» с равной вероятностью, то есть имеются 4 события (оценку «1» мы не будем рассматривать). В соответствии с таблицей 1 события несут 2 бита информации. Разумеется, удобно, когда количество событий соответствует целой степени числа 2. Если же количество событий будет 7 или 23, то появляется необходимость использования математического понятия – логарифма. С понятием «логарифм» вы познакомитесь в старших классах.



Анализ



Проанализируйте образцы заданий и их способы решения.
Поделитесь своими мыслями.

Пример №1

Ученики пошли в бассейн с 8 дорожками. Инструктор сообщил им, что они будут плавать по трем дорожкам. Какое количество информации получили ученики из данного сообщения?

Запишите краткую информацию к примеру по следующему образцу:

Дано:
 $N = 8$

Найти: $X - ?$

Решение:

$$N = 2^x$$

$$8 = 2^x \Rightarrow X = 3$$

Примечание. Мы должны ответить на вопрос: в какой степени число 2 равно 8?

Ответ: 3 бита.

Пример №2

«Тимур живет на третьем этаже многоэтажного дома». Это сообщение несет 4 бита информации. Сколько этажей в доме Тимура?

Дано:
 $X = 4$

Найти: $N - ?$

Решение:

$$N = 2^x \Rightarrow N = 2^4 \Rightarrow N = 16$$

Примечание. В данном случае на ответ не влияет, на каком этаже живет Тимур. Даже если Тимур будет жить на 7 этаже, ответ останется таким же.

Ответ: 16 этажей.

Пример №3

В школьной библиотеке книги расположены на 16 стеллажах, в каждом из которых по 8 полок. Библиотекарь сообщила Диане, что книга, которая ей нужна, находится на пятой полке седьмого стеллажа. Какое количество информации получила ученица из сообщения?

Дано:

$$N_1 = 16$$

$$N_2 = 8$$

Найти: $X - ?$ **Решение:**

$$N_1 = 2^{X_1}; N_2 = 2^{X_2};$$

$$\text{Если } 16 = 2^{X_1}, X_1 = 4;$$

$$\text{Если } 8 = 2^{X_2}, X_2 = 3.$$

Общая информация, воспринятая ученицей:

$$X = X_1 + X_2 \quad X = 4 + 3 = 7$$

Ответ: 7 бит.**Примечание.** Для получения результата достаточно знать количество стеллажей и полок.**Вопросы для размышления**

1. Чем определяется объем знаний человека, получающего сообщение?
2. Что такое вероятностный подход к измерению объема информации?
3. Чему равно сообщение, уменьшающее неопределенность знаний в два раза?
4. Что характеризует формула Ральфа Хартли?
5. Что такое случайное событие?

**Задание**

1. Поиск числа из промежутка $1... N$ дает 9 бит информации. Чему равно число N ?
2. Известно, что сообщение «Анна потеряла Букварь» несет 4 бита информации. Сколько учебников у Анны?
3. Какое количество информации необходимо для кодирования координат поля шахматной доски размером 8×8 ?
4. Было принято сообщение размером 10 бит. Какое количество информации можно получить из данного сообщения?
5. В коробке имеется 16 разноцветных шаров. Какое количество информации несет сообщение о том, что из коробки взят белый шар?
6. Чему равен объем сообщения о событии с вероятностью, равной 64 ?
7. Сообщение о том, что книга, интересующая ученика, расположена на пятой полке стеллажа, несет 3 бита информации. Сколько полок на стеллаже?
8. В коробке имеется 40 шаров, из которых 8 белых и 32 синих. Какое количество информации несет сообщение о том, что взят белый шар? Синий шар?
9. Поиск числа из промежутка $1... N$ несет 6 бит информации. Чему равно число N ?

**Домашнее задание**

1. Получена информация объемом 8 бит. Какого размера информацию можно составить из этих данных?
2. В школьной библиотеке число стеллажей несет 5 бит информации, а число полок на стеллаже несет 4 бита информации. Какое количество стеллажей и полок на каждом стеллаже в школьной библиотеке?

Раздел I Технические характеристики компьютера и сетей

1.3

Внутреннее устройство компьютера и периферийные устройства



Каково внутреннее устройство персонального компьютера?



Подумай

- Что такое бионика?
- Какое отношение она имеет к компьютеру?
- Какие идеи взяты за основу при разработке архитектуры компьютера?

Бионика (от др.-греч. *βίον* – «живущее») – прикладная наука о применении в технических устройствах и системах принципов организации, свойств, функций и структур живой природы, то есть живые формы в природе и их промышленные аналоги.



Новые знания

Внутреннее устройство компьютера

Персональный компьютер – универсальная техническая система. Его конфигурацию (основная конфигурация: системный блок, монитор, клавиатура) можно изменять по своему усмотрению.

Системный блок считается основным устройством. Устройства, расположенные внутри системного блока, называются внутренними, а устройства, присоединяемые к нему снаружи, называются внешними, или периферийными.

Внутреннее устройство системного блока:

- материнская плата;
- процессор или микропроцессор;
- оперативная память (ОЗУ);
- жесткий диск;
- привод оптических дисков (CD и DVD);
- видеографический адаптер;



внутреннее устройство компьютера – компьютердің ішкі құрылысы –
internal computer structure

материнская плата – аналық тақша – motherboard
слот расширения – слот кеңейтілімі – expansion slot

- звуковая карта;
- сетевая плата;
- порты ввода-вывода;
- блок питания.



Материнская плата

Материнская плата (англ. *motherboard*) – это основа компьютера, к которой подключаются все его остальные элементы. Она представляет собой печатную плату, на которой установлены различные микросхемы, элементы и разъемы, и служит посредником при взаимодействии различных узлов компьютера (рис. 1).



Рис. 1. Материнская плата

На материнской плате располагаются основные элементы: процессор, оперативная память, набор управляющих микросхем (чипсет), BIOS (Basic Input Output System – базовая система ввода-вывода), кэш-память, шины, слоты расширения и другие устройства. На материнской плате имеются разъемы для параллельных и последовательных портов (для подключения клавиатуры и мыши), источника питания, встроенного динамика, индикаторов и кнопок, находящихся на передней панели системного блока.

Процессор – это электронная микросхема, состоящая из множества полупроводниковых элементов (кристаллов), управляющая работой каждого устройства компьютера, а также выполняющая все вычисления и обработку информации. Полупроводниковый кристалл выполняется из особо чистого кремния. Для его изготовления используют высокоточные технологии.



На одном кристалле располагаются сотни тысяч взаимосвязанных электронных молекул, выполняющих сложные преобразования информации. Разрядность процессора определяет, какое количество бит информации он примет и обработает через свои регистры за один такт. Количество элементарных операций (тактов), выполняемых процессором за единицу времени, определяется тактовой частотой. Единицы измерения тактовой частоты – герц (Гц), мегагерц (МГц), гигагерц (ГГц).

Блок питания – вторичный источник электропитания, предназначенный для снабжения узлов компьютера электроэнергией постоянного тока путем преобразования сетевого напряжения до требуемых значений. Как правило, компьютер использует 5 или 12 вольт низкого напряжения. Различие блоков питания зависит от мощности и стабилизации напряжения. Разноцветные провода определяют величину напряжения: желтый +12 В, красный +5 В, черный 0. Мощность БП от 150 ватт до 320 ватт. В блоке питания расположен вентилятор, который охлаждает его и другие устройства, расположенные в системном блоке.



Шина в архитектуре компьютера – подсистема, служащая для передачи данных между его функциональными блоками. Имеются три основных вида шин: шина данных, адресная шина и командная шина. Шина данных – часть общей шины, предназначенная для параллельной передачи данных двоичного разряда между устройствами компьютера. Шина адреса представляет собой набор проводников, по которым передается адрес ячейки памяти, в которую или из которой пересылаются данные. Типы шин: ISA, MCA, EISA, SCSI, VLB, PCI, AGP, USB и т.д.



Чипсет (англ. chipset) – набор микросхем, спроектированных для совместной работы с целью выполнения набора заданных функций. Обеспечивает взаимодействие центрального процессора с различными типами памяти, устройствами ввода-вывода, контроллерами и адаптерами периферийных устройств.



Микросхема BIOS считается одним из основных устройств компьютера и располагается на материнской плате. Микросхема BIOS предназначена для установления параметров системы.



BIOS выполняет следующие функции:

- тестовая система POST проверяет после включения компьютера его основные компоненты;
- настраивает параметры аппаратных устройств;
- обеспечивает загрузку операционной системы с диска;
- обеспечивает взаимодействие системных и аппаратных компонентов.

В настоящее время существует множество видов BIOS. Самыми распространенными из них являются Award BIOS, AMIBIOS, Phoenix BIOS и UEFI BIOS.

До 2000 года в большинстве случаев применяли Award BIOS. Это довольно старая система, которая использовалась на компьютерах с процессорами Pentium II, Pentium III. BIOS этого типа была очень надежно защищена, так как в случае ее «поломки» невозможно было перезаписать и повторно запустить компьютер. На современных компьютерах возможность перезаписи BIOS существует, поэтому перешли на систему Phoenix BIOS.

UEFI (Unified Extensible Firmware Interface, унифицированный расширенный интерфейс) – самый последний вариант BIOS (рис. 2). Базовая система ввода-вывода определяет ход процесса загрузки компьютера, когда происходит загрузка операционной системы. А дальнейшая работа происходит под управлением ОС, обеспечивается взаимодействие различных устройств и всех комплектующих компьютера между собой.



Рис. 2. UEFI BIOS

Сопроцессор – специализированный процессор, расширяющий возможности центрального процессора компьютерной системы. У сопроцессора нет своей программы, он расширяет систему команд и регистров основного процессора, вместе с ним выполняет работу. Он не управляет системой, только ждет команду от центрального процессора. Сопроцессор дает возможность сократить выполнение арифметических операций на 80% и более, ускорить обработку фрагментов видеоизображений.

Слот расширения – разъем на материнской плате, соединенной с системной шиной, который позволяет подключать к нему дополнительные платы, расширяющие конфигурацию устройства. Через слоты обычно подключаются видеокарты, звуковые карты, сетевые карты.



Слоты расширения компьютера:

- AGP – видеокарта;
- AMR – модем, звуковая карта;
- CNR – модем, сетевая карта, звуковая карта;
- EISA – SCSI, сетевая карта, видеокарта;
- ISA – сетевая карта, звуковая карта, видеокарта;
- PCI – сетевая карта, звуковая карта, SCSI;
- SCSI, PCIe – видеокарта, сетевая карта, звуковая карта и др.

Дополнительные карты, подключаемые к материнской плате

Видеокарта – это устройство, предназначенное для вывода на экран монитора графической информации. При помощи видеокарты данные, передаваемые компьютером, преобразовываются в видеосигнал и передаются на монитор. Видеопамять Video RAM или VRAM – это внутренняя оперативная память для хранения данных, которые используются для формирования изображения на экране монитора. Чем больше объем памяти видеокарты, тем больше данных в ней может храниться.



Звуковая карта дает возможность создавать на компьютере высококачественный звук и записывать его. Карта предоставляет расширенные возможности работы со звуком, например, в компьютерных играх и других программах.



Сетевая карта дает возможность соединения компьютера с компьютерной сетью. Имеется несколько видов сетевых карт: **Ethernet**, **Token Ring** и платы, предназначенные для подключения к беспроводной сети (WI-FI).



Порты ввода-вывода – разъемы на передней или задней панели корпуса. К ним с помощью кабеля подсоединяются различные устройства. Виды и количество устройств зависят от вида и количества портов.





Периферийные устройства персонального компьютера



Рис. 3. Периферийные устройства компьютера



Анализ



Выполните анализ рисунка 3.

- Определите названия периферийных устройств компьютера.
- Выясните функции этих устройств и опишите, к каким дополнительным платам они подсоединяются.
- Определите, какие еще устройства, кроме периферийных, могут использоваться при работе с компьютером.



Синтез



Составьте общие правила использования периферийных устройств компьютера. Поясните каждый пункт составленных правил. Почему именно так составлено данное правило?



Оценка



Какова роль материнской платы в устройстве компьютера?

**Вопросы**

1. Назовите внутренние устройства компьютера.
2. Что такое материнская плата?
3. Какие устройства подсоединяются к материнской плате?
4. Какие функции выполняют периферийные устройства?
5. Что такое процессор?
6. Каково назначение графического адаптера памяти?
7. Какие дополнительные платы есть в компьютере?
8. Какие устройства подсоединяются к портам ввода-вывода системного блока?

**Домашнее задание**

№	Тема исследования	Направления и вопросы для исследования
1	История развития принтеров	<ul style="list-style-type: none"> • Назовите устройства, которыми в прошлом пользовались люди для записи и печати. • Анализ современных печатающих устройств. • Какие виды принтеров вам известны и как они работают? • Что вы знаете о 3D-принтерах?
2	Я и интернет	<ul style="list-style-type: none"> • Возникновение интернета и процесс его развития. • Устройства, необходимые для подключения компьютера к интернету, и их виды. • Пути развития и варианты использования интернета в Казахстане.
3	Развитие USB-устройств	<ul style="list-style-type: none"> • Как появились USB-порты? • В каком направлении USB-порты будут развиваться в дальнейшем?
4	Что такое архитектура компьютера?	<ul style="list-style-type: none"> • Исследования архитектуры современных компьютеров.

Раздел I Технические характеристики компьютера и сетей

1.4 Процессор и его характеристики



Что вы знаете о процессоре компьютера и его основных характеристиках?



Подумай

- В чем сходство между деятельностью человека и работой компьютера?
- Какие действия человека может выполнить компьютер?
- Может ли компьютер полностью заменить человека?
- Почему компьютер сравнивают с мозгом человека?



Новые знания

Головной мозг человека является одним из отделов центральной нервной системы. Он состоит из различных частей или структур, каждая из которых имеет свою функцию (рис. 1). Мозг часто сравнивают с другой сложной системой: цифровым компьютером.



Рис 1. Функциональные зоны коры головного мозга, отвечающие за восприятие, обработку, хранение информации



привод оптических дисков – оптикалық дискжетектер – odd optical disk drive
порты ввода-вывода – енгізу-шығару порттары – ports input-output

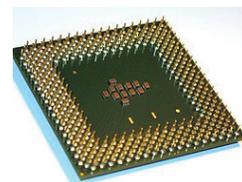
Компьютер – это электронное устройство для обработки информации с простым «мозгом», работающим по заданной программе.

Действительно, есть определенное сходство (табл. 1). Даже если учитывать, что детали компьютера можно менять и совершенствовать, мозг человека все равно не имеет аналогов в мире искусственных вычислительных систем. Максимальная оперативная память компьютера сейчас составляет 128 Гбайт, а максимальный объем жесткого диска вмещает до 14 Тбайт. Память человеческого мозга составляет 2,5 Пбайт (2500 Тбайт), что в 180 раз больше памяти компьютера.

Таблица 1. Сравнительная характеристика

Функция	Человек	Компьютер	Устройства
Хранение информации	Память	Устройства памяти	Постоянная, оперативная, кэш-память и внешняя память
Обработка информации	Система мышления	Процессор	Процессор
Восприятие информации	Органы чувств	Устройства ввода	Клавиатура, микрофон, сканер, камера, мышь
Передача информации	Речь, органы движения	Устройства вывода	Звуковые колонки, принтер, плоттер

Центральный процессор (англ. central processing unit, CPU – центральное обрабатывающее устройство) – это «мозг» компьютера, его основная микросхема. Процессор управляет работой каждого устройства компьютера и разрешает выполнение программного кода. Быстродействие компьютера напрямую зависит от скорости процессора. Работа процессора заключается в выполнении и выборе из памяти команд и данных в определенной последовательности. Это и является основой выполнения программ.



Центральный процессор вычисляет и обрабатывает данные, а также является самой дорогой микросхемой компьютера. Основные производители процессоров – компании Intel и AMD. Двухъядерный процессор производителя Intel называется Core 2 Duo, а производителя AMD – Dual Core.

Чем выше частота процессора, тем больше тепла он выделяет. Эта проблема была решена размещением в одном корпусе двух процессоров. Такие процессоры называют многоядерными. В настоящее время лидерами рынка являются процессоры фирмы Intel.

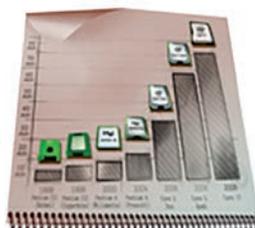
Процессоры различаются разрядностью, тактовой частотой и объемом кэш-памяти.

Основные параметры процессора

Разрядность процессора – величина, определяющая количество информации, которую центральный процессор способен обработать за один такт. Наименьший разряд единицы информации – двоичный. В мире ЭВМ один разряд называется битом (bit – часть). Если ЭВМ за один раз может обработать восемь разрядов, тогда процессор считается 8-разрядным, если 32 разряда, процессор называется 32-разрядным. Чем выше разрядность процессора, тем быстрее идет обработка данных.



Тактовая частота процессора (frequency). Важной характеристикой, влияющей на функциональность процессора, является тактовая частота. Тактовая частота – это количество операций, которые процессор способен выполнить за секунду. Она измеряется в мегагерцах. Один мегагерц – это миллион тактов в секунду. 100 МГц – сто миллионов тактов в секунду. За один такт процессор вычисляет какой-нибудь фрагмент. Поэтому чем выше тактовая частота, тем быстрее процессор обрабатывает поступившие данные.



Кэш-память процессора. Данные, обрабатываемые процессором, поступают из оперативной памяти. Как правило, процессор у себя ничего не хранит. В нем очень мало ячеек, в которых обрабатываются данные. Эти ячейки называются *регистрами*. Для облегчения работы процессора долгое время используется технология кэширования. Кэш – это промежуточный буфер с быстрым доступом. Когда какие-то данные обрабатываются в основной памяти, их копия попадает в кэш-память. В случае повторного использования этих данных их сразу можно извлечь из буфера. Если какие-либо данные нужны процессору, он в первую очередь проверяет регистры. Если нужных данных в регистрах нет, то процессор проверяет кэш-память. Если этих данных нет в кэш-памяти, процессор проверяет оперативную память. Если же и в оперативной памяти нет нужных данных, процессор делает запрос HDD-диску.



Устройство процессора

Количество ядер процессора

Процессор является техническим устройством, играющим важную роль в нашей жизни. Люди используют его в электронных устройствах в медицине, образовании, военном деле и других отраслях. Процессоры используются не только в компьютерах, но и в телевизорах, телефонах, холодильниках. Развитие микропроцессоров расширило функциональные и вычислительные возможности электронных устройств. Современные процессоры могут содержать два (двухъядерные) и более (многоядерные) вычислительных ядра. Первоначально двухъядерные процессоры имели два независимых ядра. Каждое процессорное ядро представляет собой отдельный процессор с кэш-памятью первого и второго уровней. Самым производительным считается компьютер для игр, оснащенный процессором с четырьмя ядрами (рис. 2).



Рис. 2. 4-ядерный процессор

В настоящее время центр ядра составляет кэш-память второго уровня, она еще больше увеличивает производительность. У процессора появляется возможность одновременно выполнять несколько текущих команд. По-другому можно сказать, что **многоядерный процессор** – это множество одноядерных процессоров. На материнской плате они объединяются и взаимосвязываются. Производительность многоядерных процессоров увеличивается за счет того, что данные разделяются между несколькими ядрами для обработки. На рис. 4 представлена простейшая схема ядра процессора.

Американская компания Intel впервые выпустила 8-ядерный процессор Intel Core i7-5960X Extreme Edition (рис. 3). В процессоре размещен 1 млрд транзисторов.



Рис. 3. 8-ядерный процессор

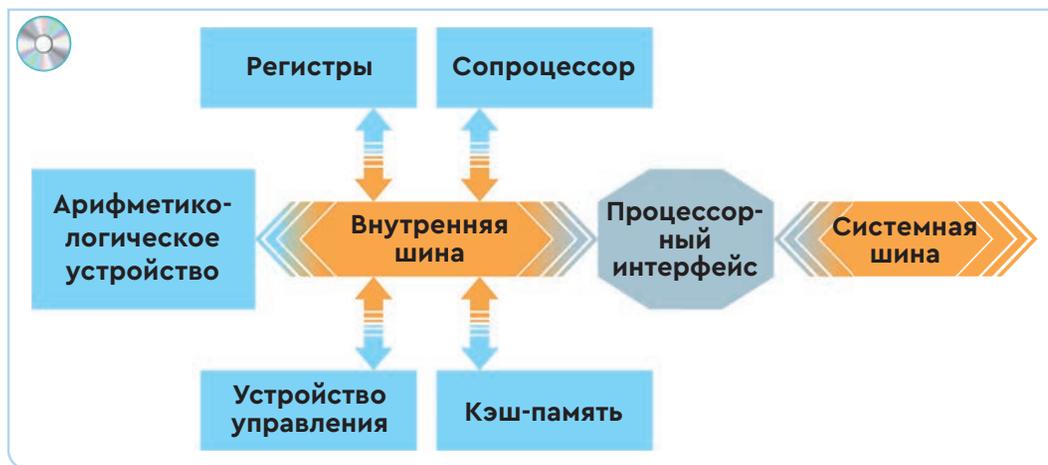


Рис. 4. Простейшее устройство ядра процессора

Арифметико-логическое устройство выполняет все математические и логические операции.

Устройство управления обеспечивает последовательность выполнения команд процессора.

Кэш-память – быстродействующая память малого объема, используемая процессором для временного хранения информации. Кэш используется в качестве буфера для ускорения обмена данными между процессором и оперативной памятью. Значения из кэш-памяти извлекаются напрямую, без обращения к основной памяти.

Регистры – ячейки памяти, расположенные внутри процессора. Это устройство временного хранения данных, числа или команды используется с целью облегчения арифметических, логических и пересылочных операций.

Сопроцессор – устройство, расширяющее вычислительные возможности основного процессора и предназначенное для обработки чисел с плавающей запятой.

Процессорный интерфейс – разъем на материнской плате, который предназначен для размещения процессора. Каждый разъем материнской платы предназначен для определенной группы процессоров.

Системная шина – информационная магистраль, обеспечивающая связь всех устройств компьютера.



Анализ



Проведите параллель между работой процессора и мозгом человека.



Синтез



«Компьютер внутри нас». Как информация влияет на настроение человека? Какие эмоции она вызывает? Приведите примеры.



Оценка



Оцените значимость процессора в организации работы компьютера.



Вопросы для размышления

1. Что такое процессор?
2. Каковы его функции?
3. Какими параметрами различаются процессоры?
4. Опишите устройство процессора.
5. Что такое многоядерный процессор?



Домашнее задание

№	Тема исследования	Направления и вопросы для исследования
1	Central Processor Unit	<ul style="list-style-type: none"> • История появления первых процессоров и их развития. • Какие фирмы, выпускающие процессоры, являются самыми известными? • На каких принципах основывается работа процессора?
2	Бионика	<ul style="list-style-type: none"> • Что такое бионика? • Становление бионики как науки. • Взаимосвязь природы и техники. • Какую пользу человечеству приносит бионика и как она будет развиваться в будущем?
3	Схожесть жизнедеятельности живых организмов с техникой	<ul style="list-style-type: none"> • Копирование, моделирование элементов живой природы. • Заимствование технологий и идей из живой природы и применение их в технике. • Исследование живой природы: птиц и животных с точки зрения техники. Применение в быту и технике принципов и структур живой природы.

Раздел I Технические характеристики компьютера и сетей

1.5 Компьютерные сети



Какие вычисления производят для определения пропускной способности сети?



Подумай

- По каким основным параметрам оцениваются компьютерные сети?
- Какие устройства необходимы для организации компьютерной сети?



Новые знания

Устройства, используемые для организации компьютерной сети

Компьютерная сеть (англ. *computer network*) – совокупность компьютеров, соединенных с помощью каналов связи и средств коммутации в единую систему для обмена сообщениями и доступа пользователей к ресурсам сети.

Сети соединяются кабелями, телефонной сетью или бывают беспроводными. Самая большая сеть в мире – интернет.

Сетевой коммутатор, хаб, свитч (от англ. *switch* – «переключатель») – устройство, соединяющее несколько компьютеров в одну сеть.



Сетевая карта дает возможность для присоединения компьютера к компьютерной сети. Имеются типы карт, обеспечивающие скорость работы 10, 100, 1000 Мбит/сек.



Роутер предназначен для подключения нескольких устройств (компьютер, ноутбук, смартфон, планшет) к локальной сети с доступом в интернет. А также это устройство для подключения к беспроводной сети, с его помощью без кабеля можно подключиться к интернету.



ADSL-модем – устройство, предназначенное для подсоединения к сети Интернет через телефонную линию.



пропускная способность сети – желінің өткізу қабілеті – network bandwidth
надежность – сенімділік – reliability

Аббревиатура **ADSL** расшифровывается как *Asymmetric Digital Subscriber Line* – «Асимметричная цифровая абонентская линия».

Патч-корд (коммутационный шнур) представляет собой электрический или оптоволоконный кабель для подключения или соединения между собой электрических устройств. Обе стороны кабеля имеют соединительные разъемы, называемые коннекторами.

Облачные технологии. Облачные вычисления

В настоящее время все большую популярность набирают так называемые **облачные технологии** (англ. *cloud technology* или *cloud computing*).

Облачные вычисления – это предоставление вычислительных служб (серверов, хранилищ, баз данных, сетевого оборудования, программного обеспечения, аналитики и т.д.) через интернет.

Это и есть так называемое «облако». Компании, которые предоставляют такие вычислительные службы, называются поставщиками облачных служб. Обычно они взимают плату за облачные вычислительные службы на основе использования так же, как взимают плату за воду или электричество дома.

Чтобы лучше представить, что такое *cloud computing*, приведем простой пример. Раньше пользователь для доступа к электронной почте прибегал к определенному ПО (мессенджерам и программам), установленному на его ПК. Теперь же он просто заходит на сайт той компании, услуги электронной почты которой ему нравятся, непосредственно через браузер. Без использования посредников абонент получает платформу различных веб-сервисов, предоставляющих клиенту удаленные вычислительные мощности.

Существуют следующие модели обслуживания облачных вычислений, предоставляемых потребителю:

- программные средства – приложения провайдера, выполняемые в облачной инфраструктуре;
- средства для развертывания в облачной инфраструктуре приложений, которые потребитель разрабатывает с использованием поддерживаемых провайдером инструментов и языков программирования;
- средства обработки данных, хранения, сетей и других базовых вычислительных ресурсов, на которых потребитель может развертывать и выполнять произвольное программное обеспечение, включая операционные системы и приложения.

Пропускная способность сети

Эффективность работы компьютерной сети определяется следующими параметрами:

- **Пропускная способность компьютера** определяет скорость передачи данных в сети: какое количество битов передано за одну секунду. Единица измерения – бит в секунду (бит/с).

- **Надежность** – способность передачи точной информации без потери данных.

- **Ценность сети** – возможность подсоединения к сети компьютеров и других устройств.

Соотношение между единицами пропускной способности канала передачи информации такие же, как между единицами измерения количества информации:

$$1 \text{ байт/с} = 2^3 \text{ бит/с} = 8 \text{ бит/с};$$

$$1 \text{ Кбит/с} = 2^{10} \text{ бит/с} = 1024 \text{ бит/с};$$

$$1 \text{ Мбит/с} = 2^{10} \text{ Кбит/с} = 1024 \text{ Кбит/с};$$

$$1 \text{ Гбит/с} = 2^{10} \text{ Мбит/с} = 1024 \text{ Мбит/с}.$$

Пропускная способность сети зависит от технических возможностей и используемых кабелей (табл. 1).

Таблица 1. Технические характеристики кабелей

Тип соединения	Пропускная способность, Мбит/с	Надежность	Возможность расширения
Электрические кабели: • витая пара • коаксиальный кабель	до 100 до 10	Низкая Высокая	Простая Сложная
Телефонная линия	1–2	Низкая	Несложная
Оптоволоконный кабель	100–200	Абсолютная	Несложная

Пропускная способность любого канала связи ограничена. Эта величина зависит от свойств устройства и кабелей, используемых для организации сети. **Объем передаваемой информации Q** определяется по следующей формуле:

$$Q = q \cdot t$$

q – пропускная способность передачи данных (единица измерения бит/с)

t – время передачи



Анализ



Рассмотрите примеры вычисления пропускной способности сети. Проанализируйте образцы заданий и их способы решения. Поделитесь своими мыслями.

Пример №1

Какой объем информации можно передать за 5 секунд с пропускной способностью 20 Кбит в секунду?

Дано:

$$q = 20 \text{ Кбит/с}$$

$$t = 5 \text{ с}$$

Найти: $Q - ?$

Решение:

$$Q = q \cdot t.$$

$$1 \text{ Кбайт} = 1024 \text{ байт}$$

$$\begin{aligned} Q &= 20 \cdot 1024 \text{ бит} \cdot 5 = 20\,480 \text{ бит/с} \cdot 5 \text{ с} = \\ &= 102\,400 \text{ бит} : 8 = 12\,800 \text{ байт} : 1024 = \\ &= 12,5 \text{ Кбайт.} \end{aligned}$$

Ответ: 12,5 Кбайт.

Пример №2

За какое время можно передать файл объемом 525 Кбайт с пропускной способностью сети 33 600 бит/с?

Дано:

$$q = 33\,600 \text{ бит/с}$$

$$Q = 525 \text{ Кбайт}$$

Найти: $t - ?$

Решение:

$$Q = q \cdot t \Rightarrow t = \frac{Q}{q}$$

$$1 \text{ байт} = 8 \text{ бит}$$

$$\begin{aligned} Q &= 525 \cdot 1024 \text{ байт} = 537\,600 \text{ байт}; \\ Q &= 537\,600 \text{ байт} \cdot 8 = 4\,300\,800 \text{ бит}; \\ t &= 4\,300\,800 \text{ бит} : 33\,600 \text{ бит/с} = 128 \text{ с.} \end{aligned}$$

Ответ: 128 с.

Пример №3

Файл объемом 1687,5 Кбайт был передан за 4 минуты. Определите пропускную способность сети.

Дано:

$$t = 4 \text{ минут} = 240 \text{ с}$$

$$Q = 1\,687,5 \text{ Кбайт}$$

Найти: $q - ?$

Решение:

$$Q = q \cdot t \Rightarrow q = \frac{Q}{t}$$

$$\begin{aligned} Q &= 1\,687,5 \cdot 1024 \text{ байт} = 1\,728\,000 \text{ байт}; \\ Q &= 1\,728\,000 \text{ байт} \cdot 8 = 13\,824\,000 \text{ бит}; \\ q &= 13\,824\,000 \text{ бит} : 240 \text{ с} = 57\,600 \text{ бит/с.} \end{aligned}$$

Ответ: 57 600 бит/с.

**Вопросы для размышления**

1. Что такое компьютерная сеть?
2. Какие устройства необходимы для настройки сети?
3. Какова функция сетевого коммутатора?
4. Что такое патч-корд?
5. По какой формуле вычисляется объем передаваемой информации?
6. По каким параметрам определяется эффективность работы компьютерной сети?
7. Какова зависимость пропускной способности сети от используемых кабелей?

**Задание**

1. Данные переданы через ADSL-соединение с пропускной способностью 256 000 бит/с. С помощью этого соединения для передачи файла необходимо 16 секунд времени. Определите объем файла в Кбайтах.
2. Скорость передачи данных устройства 56 000 бит/с. За сколько времени будет передан файл объемом 210 000 байт?
3. Данные переданы через ADSL-соединение с пропускной способностью 2 621 440 бит/с. Определите, сколько секунд понадобится для передачи файла объемом 10 Мбайт с помощью этого соединения.
4. Данные переданы через ADSL-соединение с пропускной способностью 5 242 880 бит/с. Определите, сколько секунд понадобится для передачи файла объемом 1 Гбайт с помощью этого соединения.
5. Данные переданы через ADSL-соединение с пропускной способностью 128 000 бит/с. С помощью этого соединения для передачи файла необходимо 7 минут времени. Определите объем файла в Кбайтах.
6. За какое время будет передана информация объемом 225 Кбайт модемом с пропускной способностью 14 400 бит/с?
7. Данные переданы через ADSL-соединение с пропускной способностью 16 800 бит/с. Сообщение, записанное с помощью 64-символьного алфавита, состоит из 20 480 символов. Сколько времени необходимо для передачи данной информации по сети?

**Домашнее задание**

1. Книга, набранная на компьютере, состоит из 128 страниц, на каждой странице 32 строки, в каждой строке 64 символа. На компьютере используется 256-символьный алфавит. Определите пропускную способность устройства, передающего информационный объем данной книги за 2 минуты 8 секунд.
2. За какое время модем со скоростью 28 800 бит/с передает растровое цветное изображение (каждый пиксель цвета кодируется тремя байтами) размером 640 * 480 пикселей?

Раздел II Здоровье и безопасность

2.1 Негативные аспекты использования компьютера



Каково влияние электронных устройств на организм человека? Какие эффективные способы защиты от них применяются?



Подумай

- Оказывают ли бытовые приборы негативное воздействие на здоровье человека?
- Допускается ли много времени проводить за компьютером? Почему?
- Что такое электромагнитное излучение? Как оно влияет на организм человека?
- Что такое электромагнитное поле?



Новые знания

Влияние электромагнитного излучения на организм человека

Электромагнитное поле – особый вид материи, с помощью которого электрически заряженные частицы начинают взаимодействовать между собой. Работающее электрическое оборудование приводит в движение молекулы, электроны, ионы и атомы. Учитывая, что клетки человеческого организма состоят из молекул, сильное электромагнитное поле вызывает их колебания. Это приводит к негативным изменениям в организме.

В повседневной жизни люди используют различные электрические устройства. Многие приборы: телевизор, утюг, холодильник, микроволновая печь, пылесос, компьютер, телефон и другая бытовая техника – излучают электромагнитные волны. Такие волны присутствуют в каждом доме, офисе, то есть везде, где живет и работает человек.

Волны наносят очень большой вред организму. Результаты биологических и медицинских исследований показывают, что электро-



негативный аспект – жағымсыз аспекті – negative aspect
электромагнитное излучение – электромагниттік сәуле – electromagnetic radiation

магнитные волны негативно влияют на нервную систему, внутренние органы и физиологическое развитие человека. Под воздействием электромагнитных волн возникает множество заболеваний. Волны оказывают воздействие на мыслительную деятельность человека, нарушают память. Показатели излучений источников электромагнитных волн приведены в таблице 1. Предельно допустимая норма электромагнитного излучения для человека не должна превышать 0,2 мкТл (микротесла).

Таблица 1. Показатели излучений источников электромагнитных волн

Источники электромагнитных волн	Показатели излучений (мкТл)	Превышение нормы (кратно)
Компьютер	1–100	5–500
Мобильный телефон	40	200
Холодильник	1	5
Электрический чайник	10	50
Микроволновая печь	8–100	40–500
Фен	15–17	75–85
Трамвай и троллейбус	150	750
Метро	300	1500

Негативное влияние электромагнитных волн, излучаемых компьютером, на организм человека довольно велико.

Длительная и непрерывная работа за компьютером может привести к ухудшению зрения и головным болям.

Долгая работа за компьютером ведет к следующим негативным последствиям:

- снижению остроты зрения;
- искривлению позвоночника;
- психологическим нагрузкам;
- усталости и слабости;
- снижению иммунитета;
- замедлению кровообращения.

Если вы не хотите столкнуться с перечисленными негативными последствиями, следует соблюдать определенные правила работы с компьютером. Помните: очень важно уделять внимание своему здоровью.

Способы уменьшения воздействия электромагнитных излучений от экрана компьютера

В процессе работы необходимо каждый час устраивать перерыв и выполнять разминочные упражнения. Нельзя пользоваться ноутбуком, держа его на коленях. Есть вероятность попасть под воздействие сильной радиации. При работе на компьютере или ноутбуке рекомендуется надевать защитные очки для глаз.

Выбирая монитор, следует отдавать предпочтение жидкокристаллическому варианту с надписью «Low Radiation», свидетельствующей о низком уровне излучаемой радиации.

Влажная уборка помещения и наличие ионизатора послужат дополнительной защитой от электромагнитного излучения.

Не рекомендуется ставить системный блок компьютера на стол, а также сидеть около задней панели монитора. В этих местах радиация имеет наибольшую силу излучения. При установке монитора его заднюю часть следует развернуть к окну или к стене.

От компьютера и ноутбука исходит тепло, оно уменьшает количество кальция в организме пользователя. По этой причине у человека ухудшается состояние ногтей и волос.



В комнатах, где расположены компьютер, ноутбук, телевизор и другая бытовая техника, желательно вместо пластиковых окон установить деревянные. Дело в том, что пластиковые окна не пропускают наружу радиацию, излучаемую этими приборами, и она остается в помещении.

И, наконец, не забывайте выключать компьютер по окончании работы. А телефон и планшет не кладите во время сна рядом с собой.



Основные правила работы за компьютером

- Экран монитора должен располагаться на расстоянии 50–70 см от глаз.
- Верхняя часть экрана должна находиться на уровне глаз.
- Если в классе или офисе установлено несколько компьютеров, расстояние между ними должно быть не менее 1 м.
- Задняя сторона компьютера и монитора должна быть направлена к стене.
- В помещении, в котором установлен компьютер, влажная уборка должна проводиться ежедневно. Также несколько раз в день помещение необходимо проветривать.

- Клавиатуру и мышку также необходимо ежедневно протирать салфеткой, смоченной в специальном чистящем средстве. В противном случае создается благоприятная среда для размножения бактерий.
- Во время работы за компьютером следите за осанкой.
- Во время перерыва выполните несколько упражнений для глаз: перемещайте взгляд вверх-вниз, влево-вправо; сделайте круговые движения глазами; меняйте направление взгляда – посмотрите вдаль (например, в окно), потом сфокусируйте взгляд на предмете на столе.
- Не допускайте развития «синдрома компьютерного зрения», который развивается у людей, проводящих весь рабочий день за компьютером. Источник освещения не должен быть направлен прямо в глаза, а также создавать блики на экране. Работайте на устройствах с матовыми мониторами или используйте специальные антибликовые пленки. Шторы или жалюзи также дают возможность регулировать уровень освещения.
- Если за компьютером сидит школьник, рекомендуется после 20 минут работы устраивать перерыв не менее 15 минут. Для взрослых пользователей рекомендуются ежечасные 10-минутные перерывы.

Аспект (от лат. *aspectus* – «взгляд») – точка зрения, с которой рассматривается какое-либо явление, понятие, перспектива.



Анализ



Выберите три электрических устройства, которыми вы пользуетесь ежедневно. Сделайте полный анализ влияния этих устройств на организм человека. Результаты анализа оформите в письменном виде.



Синтез



Составьте 10 вопросов для анкеты о негативном влиянии компьютера на организм человека.

**Оценка**

Назовите пять способов защиты организма человека от негативно-го воздействия компьютера. Приведите доказательства. Объясните причину. Напишите итоговый отчет.

**Вопросы для размышления**

1. Каково воздействие электромагнитных излучений на организм человека?
2. Какие заболевания может вызвать воздействие электромагнитных волн?
3. Каково воздействие электромагнитного излучения от экрана компьютера?
4. Как уменьшить негативные последствия для человеческого организма от использования компьютера?

**Домашнее задание**

№	Тема исследования	Направления и вопросы для исследования
1	Воздействие компьютера на здоровье человека	<ul style="list-style-type: none"> • Оказывает ли компьютер негативное воздействие на организм человека? • Сколько времени можно проводить перед компьютером? • Как уменьшить негативное воздействие компьютера на организм человека?
2	Польза или вред компьютерных игр	<ul style="list-style-type: none"> • Какое влияние на организм подростка оказывает длительная работа за компьютером? • Как уменьшить вред от компьютерных игр?
3	Влияние ежедневного использования электрических устройств на организм человека	<ul style="list-style-type: none"> • Какое воздействие оказывает электромагнитное излучение на организм человека? • Насколько опасны бытовые электроприборы для человека? • Как уменьшить воздействие электрических устройств на организм человека?

Раздел II Здоровье и безопасность

2.2 Безопасность в сети



Какие правила безопасности нужно соблюдать при работе в сети Интернет?



Подумай

- Объясните, как вы понимаете выражение «информационная безопасность».
- Какие опасности подстерегают пользователей интернета?
- Что вы знаете об обмане и мошенничестве в сети?
- Какие правила безопасности при работе в сети Интернет вы соблюдаете?



Новые знания

Информационная безопасность

В связи с развитием информационных технологий и цифровизацией экономики одним из важнейших вопросов в деятельности государственных органов, банков, частных компаний становится обеспечение информационной безопасности.

Информационная безопасность – состояние сохранности информационных ресурсов и защищенности законных прав личности и общества в информационной сфере.

Защита информации – единый комплекс взаимосвязанных мер, направленный на защиту информации от несанкционированного доступа, воровства, потери, удаления, изменения или копирования без разрешения правообладателя.

Существуют три направления информационной безопасности.

1. Доступность (оптимальность) – возможность получения информации в течение ограниченного периода времени. Доступность информации – свойство технических средств и информационных технологий обработки, обеспечивающее своевременный беспрепятственный доступ к интересующей информации.



сетевая безопасность – желідегі қауіпсіздік – network security
информационная безопасность – ақпараттық қауіпсіздік – information security

2. Целостность – существование информации в неискаженном виде, гарантия того, что при хранении или передаче информации не было произведено ее несанкционированное изменение или удаление.

3. Конфиденциальность – недопустимость несанкционированного доступа к информации, гарантия того, что конкретная информация доступна только тем, для кого она предназначена.

Система информационной безопасности имеет два направления (схема 1).

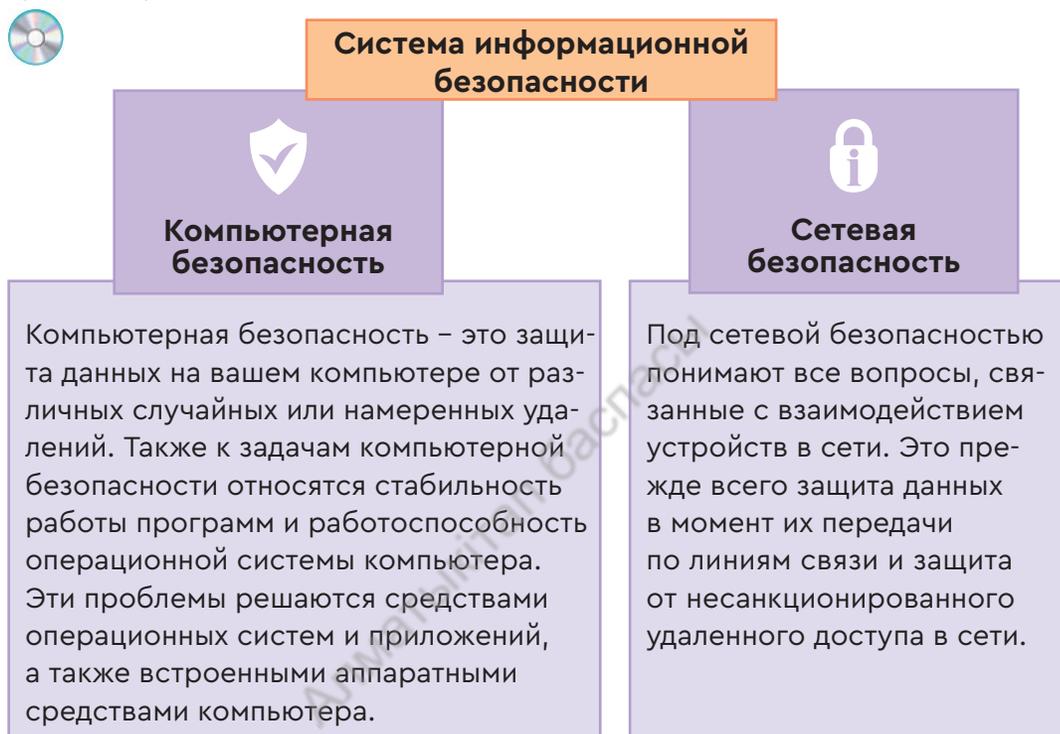


Схема 1. Система безопасности

Мошенничество в сети Интернет

Сегодня почти не осталось людей, которые не пользовались бы интернетом. У многих есть личные страницы в популярных социальных сетях и даже персональные интернет-сайты.

Получение практически любой информации и неограниченные возможности для общения сделали этот сегмент интернета самым востребованным. Сеть знакомит людей друг с другом, налаживает деловые связи, помогает обсудить широкий круг вопросов, влияет на формирование общественного мнения о событиях и явлениях.



Однако с интернетом связаны не только положительные моменты. Наряду с благородными целями сеть таит в себе ряд угроз. Встречаются пользователи с плохими намерениями или мошенники.

В виртуальной реальности, как и в обычной жизни, всегда найдутся желающие получить легкие деньги.

С какими видами мошенничества вы можете столкнуться в сети Интернет и как от них защититься? Деятельность мошенников направлена в основном на тех, кто не имеет достаточного опыта работы в сети и хочет заработать деньги.

Сегодня существует огромное количество различных способов обмана людей в интернете. Очень важно, изучая информацию на различных сайтах, не поддаваться ни на какие сомнительные и «очень выгодные» предложения быстрого заработка.

Давайте поговорим о наиболее распространенных видах обмана пользователей.

Выигрыш в лотерею или наследство от дальнего родственника.

На адрес вашей электронной почты приходит сообщение о выигрыше крупной суммы денег или письмо от юриста, который сообщает, что вы являетесь наследником недавно умершего богатого родственника.

Для получения выигрыша или наследства вам предлагают выслать деньги для оформления документов.

Волшебные кошельки. Одна из самых распространенных схем обмана в сети. Вам предлагают пополнить электронный счет на определенную сумму, обещая, что вы получите двукратную прибыль. Маленькая сумма, вложенная в проект, может вернуться к вам в удвоенном размере. Однако чем большую сумму вы переведете на указанный мошенниками счет, тем больше вероятность полной потери денег.

Предложения работы или заработка. В настоящее время это самый распространенный вид мошенничества в сети. Например, размещается объявление о вакансии наборщика текста. За работу предлагают хорошие деньги. Но для регистрации кандидата в базе данных или для открытия вашего электронного счета требуется сделать денежный перевод. Отправив указанную сумму, вы потеряете деньги и не получите работу.

Обучение заработку в интернете. Это старый и широко распространенный способ мошенничества. Вам предложат список сайтов, на которых за денежное вознаграждение вас готовы научить зарабатывать в сети. Но, переведя деньги, вы потеряете их, так как никто и не собирался заниматься вашим образованием.

Благотворительные акции. Этот вид мошенничества популярен в социальных сетях. Автор страницы сообщает, что кто-то в семье

пострадал от тяжелой болезни или несчастного случая, стал жертвой мошенников или остался без средств к существованию. Вас просят помочь деньгами.

Чтобы не стать жертвой мошенников, каждый пользователь должен соблюдать предельную осторожность при работе в интернете. Не делиться информацией, фотографиями, видеороликами о себе и своих родных с малознакомыми людьми.

Защищайте или удаляйте переписку с родственниками и близкими друзьями о своих секретах.

Что такое кибербуллинг?

Кибербуллинг (cyberbullying) – подростковый виртуальный террор, получил свое название от английского слова bull – бык, с родственными значениями: агрессивно нападать, задирать, придирааться, провоцировать, донимать, терроризировать, травить. Опасность кибербуллинга заключается в том, что в интернете (в социальных сетях, на видеопорталах) размещается ложная информация о подростке и начинается его травля. Наносится вред здоровью ребенка. В ряде случаев даже требуется помощь психолога. Как мошенничество, так и кибербуллинг в сети не остаются безнаказанными. Те, кто идет на такие преступления, не смогут избежать наказания. Хотя пользователей интернета миллиарды, злоумышленника можно вычислить. Дело в том, что каждому компьютеру при подключении к сети присваивается IP-адрес.



Что такое IP-адрес?

IP-адрес (Internet Protocol Address) – сетевой адрес узлов компьютерной сети. Провайдером (организация, которая предоставляет доступ в интернет) каждому пользователю сети выдается уникальный IP-адрес. IP-адрес состоит из четырех десятичных чисел (от 0 до 255), разделенных точками. В сети компьютер определяется по этим числам. Например, **192.168.10.43** или **128.29.15.124** (рис. 1).

По такой системе адресации миллиардам компьютеров можно задать уникальный адрес.

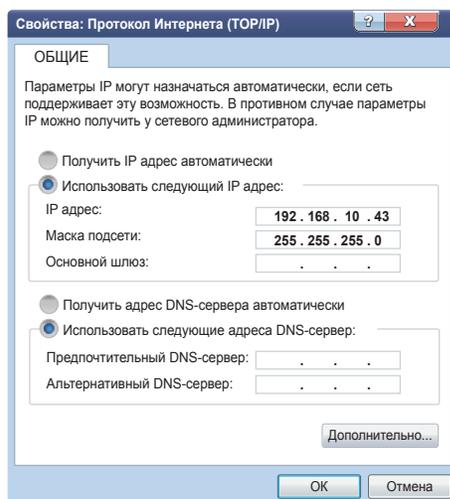


Рис. 1. Протокол интернета



Анализ



Какие виды мошенничества вы встречали в сети Интернет? Проанализируйте информацию о них. Отсортируйте по видам.



Синтез



Какую опасность для подростков представляет кибербуллинг? Как можно противодействовать ему? Разработайте правила борьбы с кибербуллингом. По каждому пункту правил приведите примеры.



Оценка



Насколько важно соблюдать безопасность при работе в сети Интернет?



Вопросы для размышления

1. Что такое информационная безопасность?
2. Почему так важна информационная безопасность?
3. Назовите направления системы информационной безопасности.
4. Что такое интернет-мошенничество?
5. Какую опасность несет кибербуллинг?
6. Можно ли поймать мошенников в сети Интернет?
7. Что такое IP-адрес?



Домашнее задание

№	Тема исследования	Направления и вопросы для исследования
1	Меры безопасности в сети Интернет	<ul style="list-style-type: none"> • Основные правила, которые должен знать пользователь сети Интернет. • Пути защиты информации в интернете.
2	Безопасное использование социальных сетей	<ul style="list-style-type: none"> • Социальные сети и их особенности. • Риски, которым пользователь подвергается в социальных сетях.
3	Интернет – это хорошо или плохо?	<ul style="list-style-type: none"> • История возникновения сети Интернет. • Полезные и вредные аспекты использования интернета. • Советы по безопасному использованию сети Интернет.

Раздел III Обработка информации в электронных таблицах

3.1

Статистические данные. Использование абсолютных и относительных ссылок



Как применять абсолютные и относительные ссылки при обработке статистических данных?



Подумай

- Что такое статистические данные?
- Какие специалисты работают со статистическими данными?



Новые знания

Что такое статистические данные?

Из истории древнего мира нам известно, что уже тогда для определения численности населения государства, количества людей, годных к воинской службе, количества скотоводческих хозяйств, площади земельных наделов (участков) выполнялись постоянные расчеты. Эти данные использовались для начисления налогов, призыва на воинскую службу и т.д. В результате изменения и развития общества возникли новые методы проведения вычислений. В середине XVII века в Англии появилось научное направление «Политическая арифметика», ориентированное на ведение учета различных изменений и анализа развития общества. Одновременно с этим в Германии появилась школа «Обязательная статистика». Благодаря таким шагам возникло новое направление науки – **статистика**. Слово «статистика» происходит от латинского status – «состояние дел». В настоящее время статистика является самостоятельной общественной наукой.

В процессе исследования развития общества получен значительный объем информации. Эту информацию необходимо было собрать, обработать, упорядочить и оформить. На примере выполнения этих работ рассмотрим применение табличного процессора MS Excel.

Использование абсолютных и относительных ссылок

Ссылка – адрес ячейки, используемый в записи формулы.

В табличном процессоре самое важное действие – копирование формулы из одной ячейки в другую. В 7 классе на уроках информа-



относительная ссылка – салыстырмалы сілтеме – relative reference
абсолютная ссылка – абсолютты сілтеме – absolute reference

тики вы узнали, что ссылка (адрес ячейки) может быть *относительной, абсолютной и смешанной*. Теперь мы подробно остановимся на способах использования этих ссылок. В прикладной программе MS Excel по умолчанию используется **относительная ссылка**, в которой при копировании формулы из одной ячейки в другую адреса изменяются автоматически. При **абсолютной адресации** во время копирования формулы адрес ячейки не изменяется, потому что ячейка рассматривается как постоянная. Для обозначения абсолютной адресации используется символ \$.

Относительная ссылка. По умолчанию все ссылки в Excel относительные. Когда вы копируете ссылку из одной ячейки в другую, она автоматически изменяется относительно позиции столбца и строки новой ячейки к ячейке, из которой вы скопировали ссылку. Например, если вы скопируете формулу =A1+B1 из строки 1 в строку 2, то формула изменится на =A2+B2. Относительные ссылки полезны в том случае, когда нужно повторить один и тот же расчет на несколько столбцов и строк. Еще один пример, если в ячейке B2 формула будет содержать ссылку на ячейку B3, расположенную на одну строку ниже. Тогда при копировании данной формулы достаточно использовать относительную ссылку. Например, A9, D25, F5 являются относительными адресами (ссылками).

Абсолютная ссылка. Абсолютные ссылки используются в противоположной ситуации, то есть когда ссылка на ячейку должна остаться неизменной при заполнении или копировании ячеек. Абсолютная ссылка обозначается знаком \$ в координатах строки и столбца, например \$A\$1. Знак доллара фиксирует ссылку на данную ячейку, так что она остается неизменной независимо от того, куда смещается формула. Другими словами, использование \$ в ссылках ячейках позволяет скопировать формулу в Excel без изменения ссылок. При абсолютной адресации знак \$ ставится как перед именем столбца, так и перед номером строки. Например, \$A\$16. Адреса \$A9, A\$7 содержат знак \$ или перед именем столбца, или перед номером строки. Такие адреса называются **смешанными**. Рассмотрим примеры использования ссылок для закрепления полученных знаний (*табл. 1*).

Таблица 1. Примеры использования ссылок

Виды ссылок на ячейку	Ссылка на следующую ячейку	
	Заполнение вправо	Заполнение вниз
A1 относительная	B1	A2
\$A1 (по столбцу абсолютная) смешанный адрес	\$A1	\$A2
A\$1 (по строке абсолютная) смешанный адрес	B\$1	A\$1
\$A\$1 абсолютная	\$A\$1	\$A\$1

Рассмотрите фрагмент электронной таблицы, заполненной числами и формулами (*табл. 2*). Обратите внимание, как меняется формула при использовании ссылок.

Таблица 2. Использование абсолютной и относительной ссылки

	Абсолютная ссылка		Относительная ссылка	
№	A	B	C	D
1	5	=A1*\$B\$2	=A1*B1	
2		8		=B2*C2
3				

В ячейке **B1** размещена формула $=A1*\$B\2 . После набора формулы, нажав клавишу **Enter**, получим **результат 40**. Здесь число, расположенное в ячейке **A1**, будет умножено на число из ячейки **B2**, ответ разместится в ячейке **B1**. Теперь скопируем эту формулу в ячейку **C2**. Какие изменения произошли? Для получения ответа на данный вопрос проанализируем ссылки, используемые в формуле.

На ячейку **A1** ссылка относительная, то есть при изменении местоположения формулы изменится и данная ссылка.

На ячейку **B2** ссылка абсолютная, поэтому при копировании формулы всегда будут использоваться данные из этой ячейки.

При копировании в ячейку **C2** формула изменит свой вид на $=B2*\$B\2 (*табл. 3*). Обратите внимание, что ссылка на ячейку **A1** изменилась на **B2**: при переходе на следующую строку она изменилась на **A2**; сместившись на один столбец вправо, приняла вид **B2**. При нажатии на клавишу **Enter** получим **ответ 64**.

Таблица 3. Копирование в ячейки формулы

	A	B	C	D
1	5	=A1*\$B\$2		
2		8	=B2*\$B\$2	
3				
4				
5				

В чем причина? При перемещении формулы вправо на одну ячейку адрес **A1** изменится на **B1**. При нажатии на клавишу **Enter** получим результат. При копировании этой формулы в ячейку **D3** формула примет вид $=C3*\$B\2 , и результат будет равен **0**, потому что в ячейке **C3** пусто и ее значение равно нулю.

Какой результат мы получим, если формулу из ячейки B1 скопируем в ячейку A4 (табл. 4)?

Таблица 4. Получение формулы при помощи копирования ячейки

	A	B	C	D
1	5	=A1*\$B\$2		
2			8 = B2*\$B\$2	
3				=C3*\$B\$2
4	=A1*\$B2			
5				

Теперь рассмотрим примеры, встречающиеся на практике.



Практическая работа

Пример. В таблице представлены данные о заработной плате сотрудников. Каким будет размер зарплаты при ее увеличении на 10%?

При повышении заработной платы на 10% надо зарплату умножить на коэффициент 1,1 ($100\% + 10\% = 110\% / 100\% = 1,1$). На рисунке 1 формула расчета зарплаты задана в ячейке D4. Если эту формулу скопировать в ячейки D5:D10, расчет будет неверным. Потому что при копировании вместо адреса D2 будет указан адрес D3 и т.д., что приведет к ошибке, как показано на рисунке 2.

СУММ				\times	\checkmark	f_x	=C4*D2	
A	B	C	D					
1	Зарботная плата сотрудников							
2	Процент повышения зарплаты 10%				1,1			
3	№	Сотрудники	Зарботная плата	Повышенная зарплата				
4	1	Сотрудник №1	100000	=C4*D2				
5	2	Сотрудник №2	120000					
6	3	Сотрудник №3	110000					
7	4	Сотрудник №4	150000					
8	5	Сотрудник №5	130000					
9	6	Сотрудник №6	80000					
10	7	Сотрудник №7	105000					

Рис. 1. Зарплата сотрудников

A	B	C	D	
1	Зарботная плата сотрудников			
2	Процент повышения зарплаты 10%			1,1
3	№	Сотрудники	Зарботная плата	Повышенная зарплата
4	1	Сотрудник №1	100000	110000
5	2	Сотрудник №2	120000	#ЗНАЧ!
6	3	Сотрудник №3	110000	12100000000
7	4	Сотрудник №4	150000	#ЗНАЧ!
8	5	Сотрудник №5	130000	1,573E+15
9	6	Сотрудник №6	80000	#ЗНАЧ!
10	7	Сотрудник №7	105000	1,65165E+20

Рис. 2. Ошибка расчета зарплаты

Для правильного автоматического вычисления заработной платы сотрудников в формуле в ссылке D2 зададим абсолютную адресацию (рис. 3). На рисунке 4 показано, как автоматически вычислить зарплату при ее повышении на 10%.

СУММ			
A	B	C	D
1	Зарботная плата сотрудников		
2	Процент повышения зарплаты 10%		
3	№	Сотрудники	Зарботная плата
4	1	Сотрудник №1	100000
5	2	Сотрудник №2	120000
6	3	Сотрудник №3	110000
7	4	Сотрудник №4	150000
8	5	Сотрудник №5	130000
9	6	Сотрудник №6	80000
10	7	Сотрудник №7	105000

Рис. 3. Абсолютная адресация ячеек

СУММ			
A	B	C	D
1	Зарботная плата сотрудников		
2	Процент повышения зарплаты 10%		
3	№	Сотрудники	Зарботная плата
4	1	Сотрудник №1	100000
5	2	Сотрудник №2	120000
6	3	Сотрудник №3	110000
7	4	Сотрудник №4	150000
8	5	Сотрудник №5	130000
9	6	Сотрудник №6	80000
10	7	Сотрудник №7	105000

Рис. 4. Автоматическое вычисление заработной платы



Анализ



Проанализируйте и сравните пути выполнения заданий 1 и 2. В процессе анализа найдите ответы на следующие вопросы.

1. В чем различие этих двух заданий?
2. Какие ссылки были использованы в процессе выполнения заданий?



Вопросы для размышления

1. Что такое статистические данные?
2. Для чего их собирают?
3. Что такое ссылка?
4. Что такое относительная ссылка?
5. В чем отличие абсолютной ссылки от относительной?
6. Что такое смешанная ссылка?
7. В чем польза использования ссылок при решении практических задач?



Задание



Задание №1. Составьте таблицу по образцу, выполните вычисления.

Таблица 2. Расчет общей цены товаров

	A	B	C	D	E
1	№	Наименование товара	Кол-во	Цена	Общая цена
2	1	Тетрадь	50	15	=C2*D2
3	2	Ручка	80	50	
4	3	Карандаш	25	6	
5	4	Ластик	10	0,45	
6	5	Линейка	20	8,15	

а) В ячейку E2 запишите формулу расчета общей суммы за тетради. Для этого умножьте количество тетрадей на цену за 1 штуку и получите формулу =C2*D2.

б) Протяните данную формулу для копирования по столбцу E. Отформатируйте таблицу и сохраните.

Задание №2. При расчете основной заработной платы служащего нужно коэффициент его должностного оклада умножить на величину базовой зарплаты, закрепленной в Казахстане (рис. 5). Закончите заполнение таблицы по приведенным данным.

	A	B	C	D
1				
2	Какую зарплату получают служащие?			
3				
4	Размер базовой зарплаты =			17697
5				
6	№	Список служащих	Коэффициент	Зарплата
7	1	Служащий №1	4,8	84945,6
8	2	Служащий №2	3,9	
9	3	Служащий №3		
10	4	Служащий №4		
11	5	Служащий №5		

Рис. 5. Расчет основной заработной платы

Задание №3. Создайте таблицу Пифагора для чисел от 1 до 10 (табл. 3).

Таблица 3. Таблица Пифагора

	1	2	3	4	5	.	.
1	1	2	3	.	.		
2	2	4	6	.	.		
3	3	6	9	.	.		
4		
5		
.							
.							

Раздел III Обработка информации в электронных таблицах

3.2

Использование абсолютных и относительных ссылок в решении практических задач



Как применять абсолютные или относительные ссылки при обработке данных разных типов?



Подумай

- Какие преимущества дает использование абсолютных и относительных ссылок при решении задач в MS Excel?



Новые знания

Абсолютные и относительные ссылки можно использовать не только внутри одного файла, но и при одновременной работе с несколькими файлами. Пусть имеются два файла, в одном из которых делается ссылка на другой файл. Выясним, отразятся ли изменения, внесенные в файл 1, на файле 2.

Рассмотрим пример. Пусть имеется прайс-лист с указанием стоимости товара в долларах США. С изменением курса доллара цена товара в тенге тоже будет меняться. В файле 1 указаны цены на товар в долларах и соответствующая цена в тенге (рис. 1). Здесь курс доллара по отношению к тенге записан в ячейке С2. Будет использоваться абсолютная ссылка на эту ячейку.

№	Наименование товара	Цена \$	Цена ₸
1	Товар №1	20	6400
2	Товар №2	40	12800
3	Товар №3	100	32000
4	Товар №4	250	80000
5	Товар №5	300	96000
6	Товар №6	400	128000

Рис. 1. Таблица цен на товары

При изменении курса доллара цена товара в тенге автоматически изменится. В файле 2 дано количество товаров, купленных за один квартал (рис. 2). Для изменения цены товара в файле 1 в соответствии с курсом доллара выполним следующую последовательность действий.

1. Чтобы в файле 2 цену товара №1 записать в ячейку С4, поставим в ней знак '=', в файле 1 выделим ячейку D4 с указанной



статистика – статистика – statistics
ссылка – сілтеме – reference

ценой товара №1. При нажатии клавиши Enter между файлом 2 и файлом 1 установится связь. Ссылка в ячейке С2 изменится: **='[1.xlsx] Лист1!\$D\$4'**. Теперь цена товара в файле 1 будет меняться в соответствии с курсом доллара, в файле 2 стоимость товара тоже будет меняться.

	A	B	C	D	E	F	G	H	I	J
1	Общее количество товаров, купленных за квартал									
2		Наименование товара	Цена, т	Количество купленных товаров			Сумма проданных товаров			Общая сумма
3	№			январь	февраль	март	январь	февраль	март	
4	1	Товар №1		20	30	80	0	0	0	0
5	2	Товар №2		30	20	70	0	0	0	0
6	3	Товар №3		40	50	90	0	0	0	0
7	4	Товар №4		60	40	30	0	0	0	0
8	5	Товар №5		80	50	20	0	0	0	0
9	6	Товар №6		40	50	70	0	0	0	0

Рис. 2. Общее количество товаров, купленных за квартал

2. При копировании формулы из ячейки С4 файла 2 в другие ячейки скопируется именно стоимость товара №1 из указанной ячейки. Чтобы этого не случилось, изменим ссылку (рис. 3).

	A	B	C	D	E	F	G	H	I	J
1	Общее количество товаров, купленных за квартал									
2		Наименование товара	Цена, т	Количество купленных товаров			Сумма проданных товаров			Общая сумма
3	№			январь	февраль	март	январь	февраль	март	
4	1	Товар №1	6400	20	30	80	128000	192000	512000	832000
5	2	Товар №2	6400	30	20	70	192000	128000	448000	768000
6	3	Товар №3	6400	40	50	90	256000	320000	576000	1152000
7	4	Товар №4	6400	60	40	30	384000	256000	192000	832000
8	5	Товар №5	6400	80	50	20	512000	320000	128000	960000
9	6	Товар №6	6400	40	50	70	256000	320000	448000	1024000

Рис. 3. Использование формулы в таблице

3. Чтобы у каждого товара записывалась его стоимость, в формуле из ячейки С4 **='[1.xlsx]Лист1!\$D\$4'** перед номером строки уберем знак \$, получим **='[1.xlsx]Лист1!\$D4'** (рис. 4).

	A	B	C	D	E	F	G	H	I	J
1	Общее количество товаров, купленных за квартал									
2		Наименование товара	Цена, т	Количество купленных товаров			Сумма проданных товаров			Общая сумма
3	№			январь	февраль	март	январь	февраль	март	
4	1	Товар №1	6400	20	30	80	128000	192000	512000	832000
5	2	Товар №2	12800	30	20	70	384000	256000	896000	1536000
6	3	Товар №3	32000	40	50	90	1280000	1600000	2880000	5760000
7	4	Товар №4	80000	60	40	30	4800000	3200000	2400000	10400000
8	5	Товар №5	96000	80	50	20	7680000	4800000	1920000	14400000
9	6	Товар №6	128000	40	50	70	5120000	6400000	8960000	20480000

Рис. 4. Результат использования формулы в таблице

Вот мы с вами и познакомились с принципами использования абсолютных и относительных ссылок в электронной таблице. Теперь для решения примеров попробуйте применить полученные знания на практике.



Практическая работа

Движение легкового автомобиля

Составьте таблицу движения пройденного пути легкового автомобиля (рис. 5), равномерно двигающегося со скоростью 72 км/ч, за каждые 2 секунды.

Задание

- Скорость автомобиля переведите в м/с.
- Определите длину пути за 20 секунд.
- За какое время автомобиль со скоростью 72 км/ч проедет 1 км?

Составьте электронную таблицу по следующему образцу. Для вычисления результата в первый столбец запишите время движения, во второй – пройденный путь за указанное время.

	А	В
1	Скорость автомобиля	
2	Скорость (км/ч)	72
3	Скорость (м/с)	20
4	Промежуток времени (с)	2

В 7 классе на уроках физики вы научились переводить км/ч в м/с. Запишите в ячейку В3 формулу $=B2*1000/3600$. Используя абсолютную ссылку, запишите в ячейку В6 формулу $=B\$3*A6$. В таблице задан общий путь легковой машины за 20 секунд и изменение длины пути за каждые 2 секунды.



Рис. 5. Иллюстрация к задаче

	А	В
1	Скорость автомобиля	
2	Скорость (км/час)	72
3	Скорость (м/с)	20
4	Промежуток времени (с)	2
5	Время (с)	Путь (м)
6	0	0
7	2	40
8	4	80
9	6	120
10	8	160
11	10	200
12	12	240
13	14	280
14	16	320
15	18	360
16	20	400

Рис. 6. Время и скорость легкового автомобиля

Результаты задания

1. Скорость легкового автомобиля равна 20 м/с.
2. Из таблицы на *рисунке 6* видно, что через 20 секунд путь равен 400 м.
3. 1 км пути со скоростью 72 км/ч автомобиль проезжает за 50 с. Чтобы получить ответ, необходимо записанные в столбцах **время** и **путь** методом автозаполнения протянуть до получения значения пути 1000 м. Тогда мы и узнаем время, потраченное на его прохождение.

**Вопросы для размышления**

1. Какие еще примеры на использование абсолютных и относительных ссылок вы можете привести?
2. Как можно связать два файла с помощью абсолютных и относительных ссылок?
3. Из каких шагов состоит алгоритм, связывающий два файла с помощью ссылок?

**Задание**

Задание №1. Ниже представлен фрагмент электронной таблицы в режиме формул. Если скопировать содержимое ячейки C1 в ячейку C3, какие значения получатся в ячейках C1 и C3?

	A	B	C
1	10	15	=B1 + \$A\$2
2	20	20	
3	30	25	
4	40	30	

Задание №2. Скопируйте формулу из ячейки C2, вставьте в ячейку C3. Какая формула получится в результате?

	A	B	C
1	30		
2	12	4	=\$A\$1*A2 +B2
3	23	5	
4	43	2	

Задание №3. У Рамазана большой дом, в котором 5 комнат. Измерив длину и ширину каждой комнаты, он получил таблицу и набрал ее

в табличном процессоре. Помогите Рамазану вычислить площадь и периметр каждой комнаты.

	A	B	C	D	E	F
1	№	Название комнаты	Длина (м)	Ширина (м)	Площадь (м ²)	Периметр (м)
2	1	Гостиная	5	6		
3	2	Детская	4	3		
4	3	Столовая	5	3		
5	4	Спальня	4	4		
6	5	Игровая	3	3		

Задание №4. Самат учится в 8 классе. На летних каникулах он помогал отцу откармливать скот. Отец был доволен трудолюбием Самата. Скот ежедневно прибавлял в весе 1% от предыдущего веса. Заданы первоначальный вес и 10 дней, в течение которых Самат откармливал скот.

Задание. Составьте формулу для вычисления и вычислите (рис. 7):

1. Какой вес наберет откормленный Саматом скот через 10 дней?
2. Сколько килограммов ежедневно набирает скот?

	A	B	C	D	E	F	G	H	I	J	K	L	
1													
2	Процент ежедневного прибавления веса от первоначального							1% =	0,01				
3													
4	День		1	2	3	4	5	6	7	8	9	10	
5	Первоначальный вес (кг)	280	282,8										

Рис. 7



Домашнее задание

Задание №5. В файле 1 задана стоимость товара с учетом изменений за каждый месяц одного квартала. В файле 2 дано количество проданных товаров за каждый месяц. В файле 2 вычислите сумму, вырученную от продажи товаров за каждый месяц.

Данные файла 1

	A	B
1	Название месяца	Стоимость товара (тг)
2	Апрель	100000
3	Май	105000
4	Июнь	103000

Данные файла 2

	A	B	C
1	Название месяца	Количество проданных товаров	Стоимость проданных товаров
2	Апрель	80	=
3	Май	57	
4	Июнь	68	

Раздел III Обработка информации в электронных таблицах

3.3 Встроенные функции



Как применять встроенные функции при решении задач в электронных таблицах?



Подумай

- Что такое функция?
- Какую роль она играет в математике и в жизни человека?
- Что такое встроенные функции в программе?



Новые знания

Функции. Определение понятия «функция»

В приложении Microsoft Excel в качестве функции рассматриваются вычислительные операции, проводимые по определенному алгоритму или формулам. У каждой функции имеется соответствующее имя. В приложении Excel имена функций можно набрать на клавиатуре или выбрать из библиотеки, выполнив команду **Формулы => Библиотека функций**. На панели инструментов за это отвечает кнопка f_x (**Вставить функцию**). **Мастер функций** предлагает вам готовые шаблоны, пользователю достаточно записать аргументы.

В приложении MS Excel имеется более 200 встроенных функций, предназначенных для решения простейших и сложных задач. А также пользователи приложения имеют возможность создавать необходимые формулы с помощью встроенного приложения – Visual Basic for Applications (VBA).

Для ускорения поиска функции поделены на основные категории: финансовые, логические, текстовые, дата и время, ссылки и массивы, математические, статистические, инженерные, аналитические, проверка свойств и значений (*рис. 1*).

В качестве аргументов функции могут использоваться числа, числовые или буквенно-числовые выражения, адрес ячейки, адрес диапазона.

Например:

= СУММ(A5:A9) – здесь СУММ – функция, вычисляющая сумму значений ячеек A5, A6, A7, A8, A9;



встроенные функции – кірістірілген функциялар – bult-in functions
функции – функциялар – functions

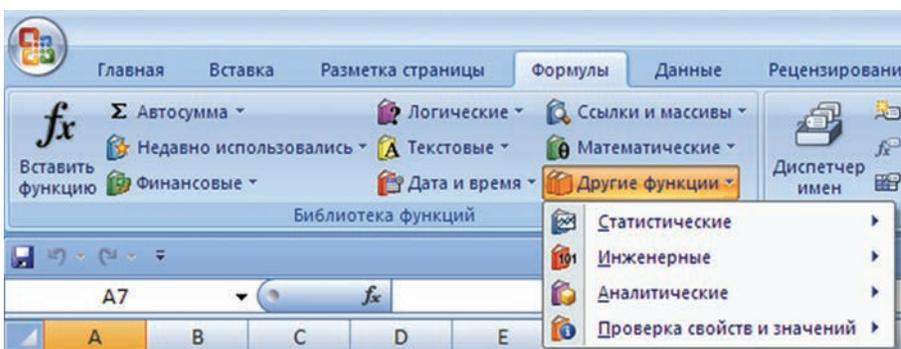


Рис. 1. Виды функций

= СРЗНАЧ(G4:G6) – здесь СРЗНАЧ – функция, вычисляющая среднее арифметическое содержимое ячеек G4, G5, G6.

При составлении сложных функций (одна функция в другой) используются круглые скобки: = ОКРУГЛ(СРЗНАЧ(Н4: Н8); 2).

Ввод аргументов функции с помощью Мастера функций

Если функция, записываемая с помощью Мастера функций, имеет аргументы, то открывается диалоговое окно, как показано на *рисунке 2*. Это окно предназначено для ввода аргументов.

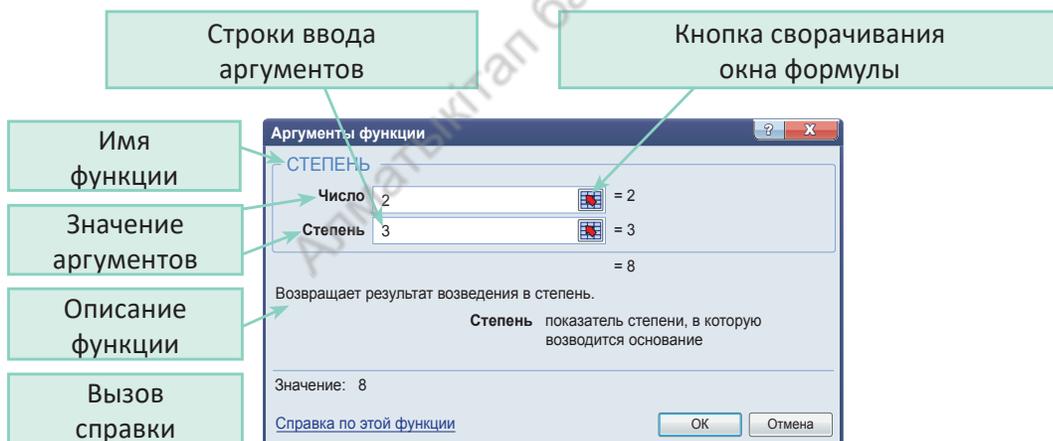


Рис. 2. Окно мастера функций

Способы ввода аргументов функции

1. Используемые адреса (ячейки или диапазон) можно набрать на клавиатуре.
2. Используемые ячейки и диапазоны можно указать, выделив их на рабочем листе.

Теперь рассмотрим функции, которые используются при выполнении вычислений чаще остальных.



Математические функции

ABS (число) – модуль числа.

SIN (число), COS (число), TAN (число) – вычисляет синус, косинус, тангенс заданного угла.

ГРАДУСЫ (угол) – переводит радианы в градусы.

ЗНАК (число) – определяет знак числа; возвращает значение 1 – если число положительное, 0 (ноль) – если число равно 0, и -1 – если число отрицательное.

КОРЕНЬ (число) – функция, вычисляющая квадратный корень заданного числа.

СУММ (число 1; число 2; ...) – вычисляет сумму всех чисел, указанных в качестве аргументов.

СТЕПЕНЬ (число; степень) – заданное число возводится в указанную степень.

Статистические функции

МАКС (аргумент 1; аргумент 2;...) – определяет максимум среди заданных аргументов.

МИН (аргумент 1; аргумент 2;...) – определяет минимум среди заданных аргументов.

СРЗНАЧ (аргумент 1; аргумент 2;...) – определяет среднее значение заданных аргументов.

СЧЕТЕСЛИ (диапазон; условие) – подсчитывает количество пустых ячеек в диапазоне, удовлетворяющих заданному условию.

Логические функции

Логические функции служат для проверки результата какого-либо условия. При составлении условий используются операторы сравнения: =, >, <, <> (не равно), >= (больше или равно), <= (меньше или равно).

ЕСЛИ (логическое_выражение; значение 1; значение 2) – значение 1 будет результатом, если логическое выражение истинно, в противном случае результатом будет значение 2. Например: в ячейке A1 записано число 30000, а в ячейке B1 – формула =ЕСЛИ(A1<20000; 12; 15). Так как условие не выполнено, то результатом будет 15.

Логические функции **И**, **ИЛИ** предназначены для составления сложных условий.

И (логическое_выражение 1; логическое_выражение 2; ...) – принимает значение **ИСТИНА**, если все аргументы принимают значение **ИСТИНА**, в противном случае принимает значение **ЛОЖЬ**.

ИЛИ (логическое_выражение 1; логическое_выражение 2; ...) – принимает значение **ИСТИНА**, если хотя бы один аргумент принимает значение **ИСТИНА**, в противном случае принимает значение **ЛОЖЬ**.

Например: **=ЕСЛИ(И(A1>=20000;A1<40000);15;18)** – если A1 принимает значение из промежутка от 20000, выключая его, до 40000, не выключая его, тогда значение функции равно 15, в противном случае – 18.

С помощью встроенной функции **ЕСЛИ** можно составлять сложные условия. Например:

=ЕСЛИ(A1<20000;12;ЕСЛИ(A1<40000;15;18)) – если в ячейке A1 находится число меньше 20000, то функция примет значение 12, если ячейка A1 содержит число меньше 40000, тогда значение функции равно 15, в противном случае функция принимает значение 18. Если приложение MS Excel не может **правильно определить формулу**, тогда в ячейке возможно наличие одной из приведенных в таблице ошибок.

Ошибка	Пояснение
# Дел/0!	В формуле содержится деление на ноль
# Знач!	Недопустимый тип аргумента
# Имя?	Ошибка в написании имени
# Н/Д	Неопределенные данные
# Пусто!	Пересечение двух областей, которые не имеют общих ячеек
#Ссылка!	Задана некорректная ссылка
# Число!	Используется аргумент, не входящий в область определения функции



Практическая работа

Исследование чисел

Задано пятизначное число, например 54289. При исследовании выполните следующие задания:

- 1) найдите квадрат числа;
- 2) найдите квадратный корень числа;
- 3) найдите сумму цифр в составе числа;
- 4) найдите среднее арифметическое цифр, составляющих число;
- 5) найдите наименьшее значение среди цифр, составляющих число.

Практическое задание выполним в приложении MS Excel в соответствии с рисунком 3.

МИН		=СРЗНАЧ(5;4;2;8;9)	
A	B	C	D
1			
2	Результаты исследования числа 54289		
3	1	Вычисление квадрата числа	2947295521
4	2	Извлечение квадратного корня из числа	233
5	3	Вычисление суммы цифр числа	28
6	4	Нахождение наименьшего из цифр, составляющих число	2
7	5	Вычисление среднего арифметического цифр числа	=СРЗНАЧ(5;4;2;8;9)

Рис. 3. Таблица выполнения задания

При выполнении практической работы будем использовать мастер функций, а также функции, приведенные ниже. В ячейки можно записать формулы, составленные с их помощью.

Ячейка C3: =СТЕПЕНЬ (54289; 2)

Ячейка C4: =КОРЕНЬ (54289) (рис. 4)

Ячейка C5: =СУММ (5; 4; 2; 8; 9)

Ячейка C6: =МИН (5; 4; 2; 8; 9)

Ячейка C7: =СРЗНАЧ (5; 4; 2; 8; 9)

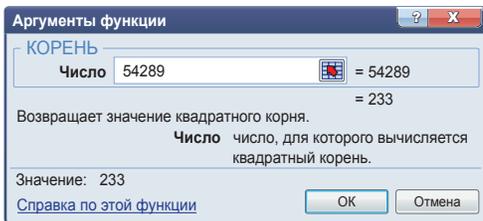


Рис. 4. Вычисление корня



Вопросы для размышления

1. Дайте математическое определение понятия «функция».
2. Назовите основные функции приложения MS Excel.
3. Какие категории функций имеются в приложении?
4. Какие существуют способы ввода аргументов функции?
5. Какую структуру имеет рабочее окно MS Excel?
6. Назовите математические функции программы MS Excel.
7. Какую роль выполняют логические функции?



Домашнее задание

Канат учится в 8 классе. В его школе 8 параллельных классов, в которых по результатам I четверти получены результаты качества знаний. Канат, изучив полученные данные, решил занести их в таблицу. Помогите ему выполнить задание.

№	Содержание задания	Ячейка для записи результатов				
			А	В	С	Д
1	Определение класса с самым высоким уровнем качества знаний	C1	8 «А»	74%		
2	Определение класса с самым низким уровнем качества знаний	C2	8 «Б»	72%		
3	Определение среднего качества знаний среди классов	B8	8 «В»	51%		
4	На сколько процентов нужно поднять уровень знаний класса с самым низким качеством знаний, чтобы достичь среднего уровня?	C3	8 «Г»	73%		
5	Разница между высоким и низким уровнем качества знаний	C4	8 «Д»	81%		
			8 «Е»	37%		
			8 «Ж»	64%		

Раздел III Обработка информации в электронных таблицах

3.4

Использование встроенных функций при решении практических задач



Какие возможности открывает использование встроенных функций при решении практических задач?



Подумай

- Как применять встроенные функции для обработки вычислительных задач, встречающихся в повседневной жизни?
- В каких изученных вами разделах алгебры можно эффективно применять встроенные функции при решении задач?



Практическая работа

Предлагаемые задания ориентированы на практическое использование встроенных функций табличного процессора. Вы научитесь строить графики функций для задач, связанных с повседневной жизнью. Например, сможете считать расходы на электроэнергию. В задачах также описана последовательность их решения.

Практическая работа №1

Похвальный лист

В конце учебного года ученики, получившие по всем предметам «отлично», награждаются похвальным листом. Рамазан учится в 8 классе. Его оценки представлены в виде электронной таблицы (рис. 1).

Задание. Определите, будет ли Рамазан в конце учебного года в числе отличников. Получит ли он похвальный лист?

Идея выполнения задания. В столбец «Анализ» ставим 1 в том случае, если оценка «5», в ином случае ставим 0. Если у ученика по всем предметам оценка «5», тогда сумма чисел столбца «Анализ» будет равняться количеству предметов. В нашем примере это 14.



ссылки и массивы – сілтемелер және жиымдар – references and arrays
математическое выражение – математикалық өрнек – mathematically expression

	A	B	C	D
1	Годовые оценки ученика			
2				
3		Предмет	Оценка	Анализ
4	1	Казахский язык и литература	5	1
5	2	Самопознание	5	1
6	3	Русский язык	4	0
7	4	Русская литература	5	1
8	5	Английский язык	4	0
9	6	История Казахстана	5	1
10	7	Алгебра	4	0
11	8	Геометрия	5	1
12	9	Информатика	5	1
13	10	Физика	5	1
14	11	Химия	3	0
15	12	Биология	5	1
16	13	География	4	0
17	14	Физическая культура	5	1
18	Средняя оценка		4,6	9
19	Получит ли похвальный лист?			Нет

Рис. 1. Итоговые оценки ученика

Алгоритм выполнения задания

1. В столбце «Анализ» в ячейку D4 запишем формулу с логической функцией: =ЕСЛИ (C4=5; 1; 0).
2. Формулу из ячейки D4 скопируем до ячейки D17 включительно.
3. В ячейку D18 запишем сумму ячеек с D4 до D17: =СУММ (D4:D17).
4. В ячейку D19 запишем логическую функцию: =ЕСЛИ (D18=14; «Да»; «Нет»).
5. В ячейку C18 запишем формулу вычисления среднего арифметического числа с ячейки C4 до ячейки C17: =СРЗНАЧ (C4:C17).

Практическая работа №2

График функции

Задание. Постройте график функции $y = x^2 - 4$. Для этого в приложении MS Excel составим таблицу, в которой каждому аргументу x соответствует ордината y (табл. 1).

Таблица 1. Таблица зависимости y от x

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	x	-3,0	-2,5	-2,0	-1,5	-1,0	-0,5	0,0	0,5	1,0	1,5	2,0	2,5	3,0
2	y	5,0	2,3	0,0	-1,8	-3,0	-3,8	-4,0	-3,8	-3,0	-1,8	0,0	2,3	5,0

Алгоритм выполнения задания

1. При построении таблицы начальное значение $x = -3$ с шагом 0,5.
2. В ячейку B2 запишите формулу =B1*B1-4 и скопируйте ее до ячейки N2 включительно.
3. Выделив таблицу зависимости y от x и выполнив последовательность команд Вставка=>Диаграмма Точечная => Точечная с гладкими кривыми, вы получите диаграмму, изображенную на рисунке 2. При изменении данных в таблице изменится и диаграмма.

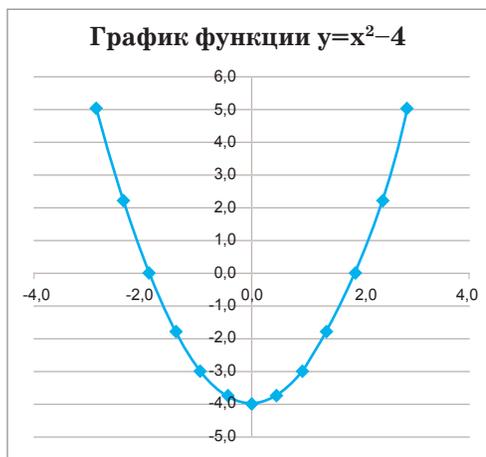


Рис. 2. График

Практическая работа №3



Вычисление расхода электроэнергии

Даны стоимость электроэнергии за 1 кВт/ч и показания электросчетчика. Также известен расход электроэнергии за прошлый и текущий месяцы.

Задание

1. Вычислить общее количество потраченной электроэнергии за прошлый и текущий месяцы.
2. Вычислить средний показатель потраченной энергии потребителями за текущий месяц.
3. Определить потребителя с наибольшим показателем использования электроэнергии.
4. Определить потребителя с наименьшим показателем использования электроэнергии.
5. Построить диаграмму потраченной электроэнергии потребителями.

Выполнив задание в приложении MS Excel, вы научитесь строить таблицы и форматировать их, вставлять в ячейки формулы, а полученные знания по работе с абсолютными ссылками, функциями и диаграммами сможете использовать на практике.

Алгоритм выполнения задания

1. Создадим таблицу по образцу, представленному на *рисунке 3*, отформатируем данные в ячейках.
2. Внесем в ячейки соответствующие данные.

	A	B	C	D	E
1	Вычисление электроэнергии				
2	стоимость 1кВт/час 15 тенге				
3	Дом №	Показатель счетчика за прошлый месяц (кВт/ч)	Показатель счетчика за текущий месяц (кВт/ч)	Расход (кВт/ч)	Цена за электроэнергию (тенге)
4	Дом №1	2243	2347	104	1560
5	Дом №2	4256	4401	145	2175
6	Дом №3	356	499	143	2145
7	Дом №4	7411	7789	378	5670
8	Дом №5	8685	8888	203	3045
9	Дом №6	25	231	206	3090
10	Дом №7	485	567	82	1230
11	Общий расход	23461	24722	1261	18915
12	Средний расход	3351,6	3531,7	180,1	2702,1
13	Наибольшее потребление			378	
14	Наименьшее потребление			82	

Рис. 3. Данные вычисления электроэнергии

3. По заданию 1 в ячейках B11, C11, D11, E11, используя функцию суммирования, разместим соответствующие данные. Например, в ячейку B11 вставим формулу =СУММ(B4:B10).
4. Для определения среднего показателя электроэнергии, потраченной в текущем месяце, запишем формулу =СРЗНАЧ(D4:D10) в ячейку D12.
5. Для определения потребителя с наибольшим показателем использования электроэнергии в ячейку D13 запишем формулу =МАКС(D4:D10). Для определения потребителя с наименьшим показателем использования электроэнергии в ячейку D14 запишем формулу =МИН(D4:D10).
6. Чтобы построить диаграмму потраченной потребителями электроэнергии, выделим данные диапазона D4:D10 и с помощью команды **Вставка** ⇒ **Диаграмма** получим новую диаграмму (рис. 4).



Рис. 4. Диаграмма



Домашнее задание

1. Заданы фамилия, имя, возраст и рост учащихся. Сколько учеников могут записаться в секцию волейбола, если набирают ребят не старше 14 лет и ростом выше 165 см?
2. В зависимости от длительности хранения продуктов на складах цена на них уменьшается. Если продукт хранится более 10 месяцев, то его цена снизится на 80 процентов, если срок хранения более 6 и менее 10 месяцев, цена упадет на 50 процентов. Составьте сводную таблицу товаров, хранящихся на торговом складе. В таблицу должны быть внесены наименования товаров, сроки хранения, первоначальная цена, цена после уценки.
3. Постройте график функции $y = -x^2 + 3$.

Примечание. Для таблицы возьмите реальные данные. Если вы напишете, например, что рост ученика составляет 300 см, то это не будет соответствовать действительности.

Раздел III Обработка информации в электронных таблицах

3.5 Анализ данных на основе имеющейся информации



Как в приложении MS Excel сделать прогноз на основе имеющейся информации?



Подумай

- Что такое прогноз?
- Какие прогнозы вы составляете в повседневной жизни?
- Существуют ли прогнозы в науке и технике?



Новые знания

В статистике **динамическими рядами** называют ряды, отражающие развитие событий, которые происходят систематизированно и в хронологическом порядке. В динамическом ряду для каждого временного отрезка приводятся два показателя: показатель времени t и уровень ряда y . Динамические ряды образуются в результате измерения некоторых показателей: технических и социальных систем, природы, экономики. С помощью динамических систем можно исследовать изменение температуры объекта, пропускную способность сети, провести анализ продажи товара за определенный промежуток времени. Динамические ряды дают возможность определить направление или тренд основного развития определенных показателей.

Анализ динамических рядов – совокупность математико-статистических методов, предназначенных для определения структуры динамических рядов и их прогнозирования.

Слово «**прогноз**» (от греч. *prognosis* – «предвидение», «предсказание») означает предсказание будущего с помощью научных методов, а также сам результат предсказания. Прогноз – это научная модель явлений или предстоящих событий.

Прогнозирование – это процесс разработки прогнозов.



анализ данных – деректерді талдау – data analysis
прогноз, прогнозирование – болжам – forecast, prediction

К методам прогнозирования относятся:

- статистический метод;
- экспертная оценка (метод Дельфи);
- моделирование.

Линия тренда в Excel – это график функции, который используется для составления прогнозов на основе имеющихся табличных данных.

Для наглядной иллюстрации тенденций изменения значений исходных табличных данных применяется линия тренда. Главной задачей линии тренда является составление прогноза дальнейшего развития событий.

Виды линий тренда в программе Excel

В программе Excel можно выбрать один из шести видов линий тренда: линейный, логарифмический, полиномиальный, степенный, скользящий средний, экспоненциальный. Рассмотрим подробнее линейный тренд.

Линейный тренд хорош при моделировании характеристик, значения которых увеличиваются или убывают с постоянной скоростью. Это наиболее простая в построении модель исследуемого процесса. Используется этот вид тренда для создания прямой линии, которая очень хорошо описывает простой линейный набор данных, когда точки данных расположены близко к прямой. Линейный тренд используется для анализа числовых данных, например, для оценки зависимости между двумя видами значений. При построении линейного тренда связи между двумя параметрами определяются коэффициентом линейной функции по методу наименьших квадратов.

Для нахождения значений коэффициента линейной функции используется коэффициент R-квадрат.

Коэффициент R-квадрат

Коэффициент R-квадрат называют величиной достоверности, с его помощью можно определить уровень точности прогноза. При значении коэффициента, равном единице (или близком к единице), можно говорить о возможности использования полученных данных для прогноза. При сопоставлении данных линии тренда в программе Excel автоматически вычисляется значение коэффициента R-квадрат. При необходимости можно эти значения отразить на диаграмме.

Как добавить линию тренда в Excel? Чтобы ответить на этот вопрос, рассмотрим следующий пример.

Канат любит читать страницы блогеров в социальных сетях. Он обратил внимание, что число посетителей страниц зависит от количества постов и репостов. Если на странице опубликовано 100 постов, то количество ее посетителей составляет 228 тысяч (табл. 1).

Теперь Каната интересуют ответы на следующие вопросы: сколько людей просмотрят страницу, если количество опубликованных постов будет равняться 120? 150?

Для определения отношения между двумя величинами Канат составил формулу, по которой, зная количество постов, можно вычислить количество посетителей страницы. Что представляет собой данная формула?

Таблица 1. Количество постов и посетителей в соцсети

№	Количество постов на странице	Количество посетителей страницы (тысяч человек)
1	30	67
2	42	88
3	57	92
4	69	120
5	75	145
6	87	184
7	95	199
8	100	228
9	120	?
10	150	??

Для составления равенства, определяющего отношение между двумя величинами, введем обозначение. Количество статей обозначим x , количество посетителей обозначим y . По таблице 1 можем определить, что отношение между этими величинами линейное. Тогда запишем формулу линейного уравнения, известную из курса алгебры:

$$y = kx + b.$$

Если в этом уравнении будут известны значения коэффициента k и свободного члена b , то, взяв любое значение x , вычислим соответствующее ему значение y .

На основе табличных данных построим линейный график.

Для этого, выделив табличные данные, необходимо выполнить следующую последовательность команд: **Вставка => Диаграммы => Точечная => Точечная с маркерами**. В результате получим диаграмму, как на *рисунке 1*.

На диаграмме по оси x расположено количество постов, по оси y – количество пользователей страницы (тысяч человек). Точки не расположены на одной линии (прямой), а разбросаны. Можем ли мы провести одну линию, максимально близко расположенную ко всем

точкам? Для ответа на данный вопрос нужно проверить корреляцию этих точек. Если при изменении одной величины изменяется среднее значение другой, то такая статистическая зависимость называется *корреляционной зависимостью*. В программе Excel имеется функция **КОРРЕЛ** (корреляция) (рис. 1). Значение корреляции функции обозначается буквой **R**. **Корреляция** в переводе с латинского *correlatio* – «отношение», «взаимосвязь». Если значение корреляции функции $R > 0$, то связь положительно возрастающая, если $R < 0$, то связь отрицательно убывающая. При приближении значения корреляции функции к 1 линии связи приближаются к прямой. В нашем случае $R = 0,96704012$, то есть отчетливо видна линейная зависимость.

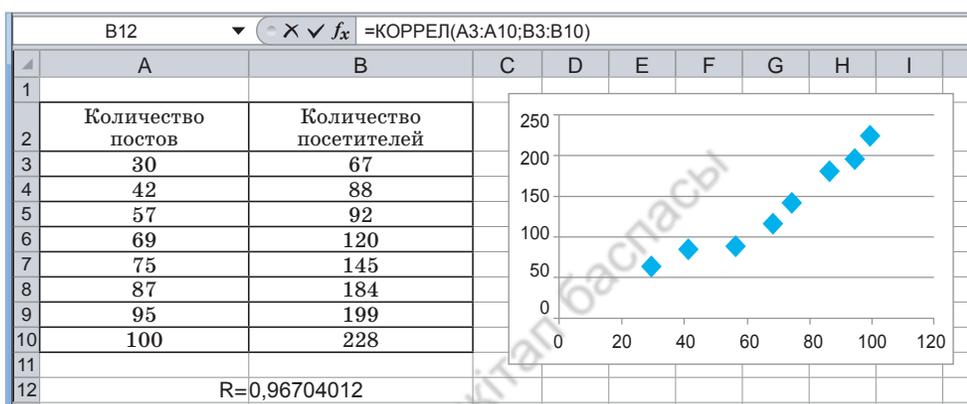


Рис. 1. Корреляция

Для изображения на диаграмме линий линейного тренда выберем одну точку, например, пятую по счету. Выделим ее левой кнопкой мыши, в контекстном меню выполним команду «Добавить линию тренда...» (рис. 2). Из шести видов, представленных на рисунке 2, выберем тренд «Линейная». Для отражения на диаграмме уравнения зависимости линейного тренда установим флажок в строке «показывать уравнение на диаграмме». Для размещения на диаграмме достоверной величины аппроксимации установим фла-

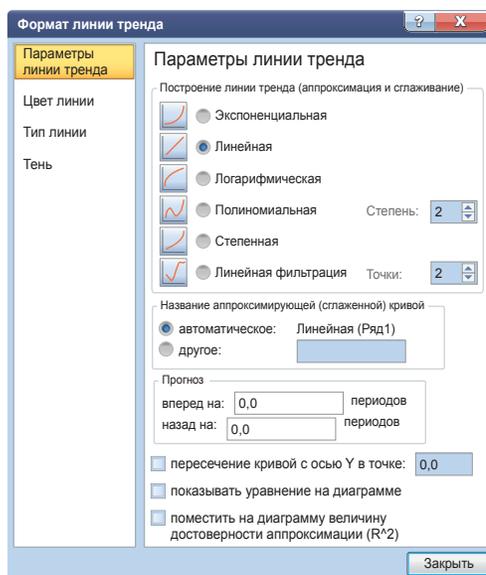


Рис. 2. Формат линии тренда

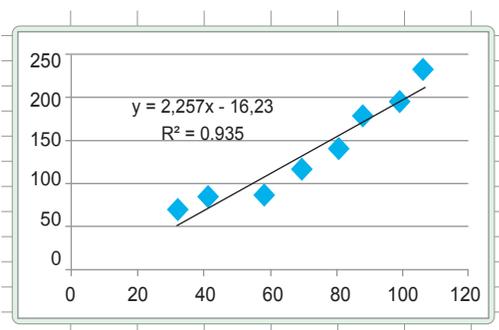


Рис. 3. Уравнение зависимости и значения R^2

жок в строке «поместить на диаграмму величину достоверности аппроксимации R^2 ». При нажатии на кнопку «Заккрыть» получим диаграмму, на которой будет отражена линия тренда, записаны уравнение зависимости и значение R^2 (рис. 3).

Теперь найдем ответы на вопросы Каната. Для этого можно использовать следующие методы.

Метод 1. На диаграмме для $x = 120$ по линейной зависимости $y = 2,257x - 16,23$ нужно выполнить прогнозирование (табл. 1). Если в результате вычислений количество посетителей страницы составит 254 тысячи, а количество статей – 150, то можно предположить, что посетителей будет 322 тысячи.

$$y = 2,257 \cdot 120 - 16,23$$

$$y = 254,31$$

Метод 2. Для вычисления прогноза используем функцию ПРЕДСКАЗ. Она относится к категории статистических. Результаты действия функции можно увидеть на рисунке 4:

= ПРЕДСКАЗ (известные_значения_y; известные_значения_x)

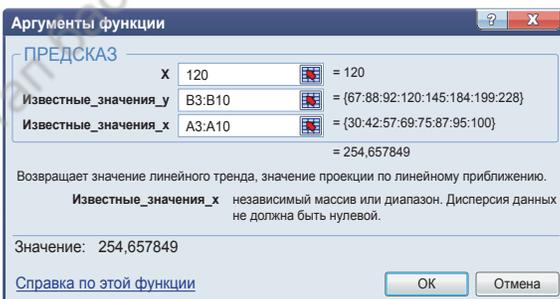


Рис. 4. Результаты функции

Вместо x из таблицы 1 возьмем для прогноза значение 120, вместо y укажем количество пользователей. Для построения прогноза вместо x возьмем промежуток значений, расположенных выше $x=120$. Результат прогнозирования получился равным 254,65, как и в первом методе.



Анализ



Проанализируйте проблему, которую решает Канат. Покажите отдельно теоретическую и практическую части прогнозирования.

**Синтез**

Составьте два примера на прогнозирование из повседневной жизни. Докажите зависимость между двумя величинами в составленных вами примерах. Например, зависимость рождаемости от экономического роста в Казахстане в ближайшие 10 лет.

**Оценка**

Какова значимость прогнозирования на основе имеющихся данных? Что дает такое прогнозирование?

**Вопросы для размышления**

1. Что такое динамические ряды? Какие функции они выполняют?
2. Какие функции используются в программе Excel для прогнозирования?
3. Объясните назначение функции КОРРЕЛ.
4. Что такое линейный тренд?
5. Какие виды линейных трендов вы знаете?
6. Каково назначение функции ПРЕДСКАЗ?

**Домашнее задание**

1. Один из составленных вами примеров используйте для прогнозирования на компьютере.
2. Имеются данные об изменении погоды за 10 дней (табл. 2). Составьте прогноз погоды на 11-й день.

Таблица 2. Погода за 10 дней

Дни	Температура за этот день
1	+2
2	+3
3	+5
4	+4
5	+6
6	+10
7	+13
8	+9
9	+15
10	+18
11	???

3.6-3.7 Решение прикладных задач



Как применять встроенные функции, различные типы и форматы данных, а также на их основе строить графики функций при решении прикладных задач в приложении MS Excel?



Подумай

- Что такое прикладная задача?
- Какое преимущество дает решение прикладных задач на компьютере в отличие от выполнения расчетов вручную?
- Какие прикладные задачи можно решить с помощью функций?



Практическая работа

Прикладные задачи – задачи профессиональной направленности, возникающие в какой-либо сфере человеческой деятельности: математике, физике, химии, экономике, бухгалтерском учете и др. Применение теоретических материалов на практике называют прикладными задачами. Многие из них можно легко и быстро решить с помощью приложения MS Excel.



Применение электронных таблиц для решения математических задач

Задание. Даны функции $y = 2x + 2$ и $y = -2x + 6$. Постройте графики этих функций и найдите точку их пересечения.

Алгоритм выполнения задания

1. При построении таблицы примем значение x в интервале от -5 до $+5$ с шагом 1 .
2. По значениям x построим график функции $y = 2x + 2$.
Для вычисления значения y в ячейку **A3** запишем формулу $y = 2 * B3 + 2$, скопируем ее в ячейки с **A3** по **A13**.
3. По значениям x построим график функции $y = -2x + 6$. Для вычисления значения y в ячейку **C3** запишем формулу $= -2 * D3 + 2$, скопируем ее в ячейки с **C3** по **C13**.



электронная таблица – электрондық кесте – spreadsheet
вставить – кірістіру – insert

4. Для построения графика функции $y = 2x + 2$ выделим данные из диапазонов **В3:В13** и **А3:А13**, выполним команду **Вставка=> Диаграммы => Точечная => Точечная с гладкими кривыми** и получим красную линию на диаграмме (рис. 1).
5. Для построения графика функции $y = -2x + 6$ на этой же диаграмме выделим данные из диапазона **С3:С13**, нажмем кнопку «**Вставить**». На диаграмме появится синяя линия. Координаты точки пересечения графиков можно увидеть в 9-й строке таблицы, изображенной на *рисунке 1*. Других таких совпадений нет.

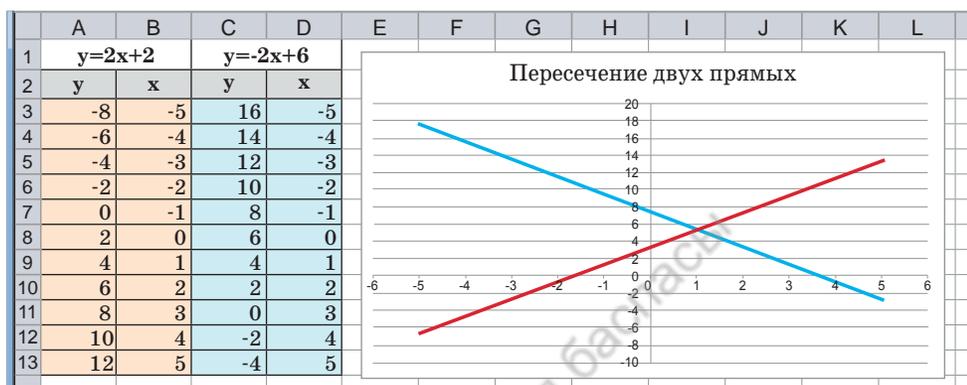


Рис. 1. Пересечение графиков

Применение электронных таблиц для решения физических задач

Задание. Постройте график зависимости силы тока от сопротивления $I(R)$. Сопротивление (R) изменяется в пределах от 1 до 10 Ом с шагом 1 Ом.

Алгоритм выполнения задания

Для выполнения задания запишем закон Ома для части цепи из курса физики. Из формулы видно, что сила тока и сопротивление обратно пропорциональны. $I = \frac{U}{R}$ В ячейку D1 запишем постоянное значение напряжения. Например, $U=40В$. Ячейку D1 будем использовать с абсолютной адресацией. В столбец В запишем значения сопротивления, а в ячейку A2 запишем формулу $=D\$1/B2$ и скопируем ее до ячейки A11 (рис. 2).

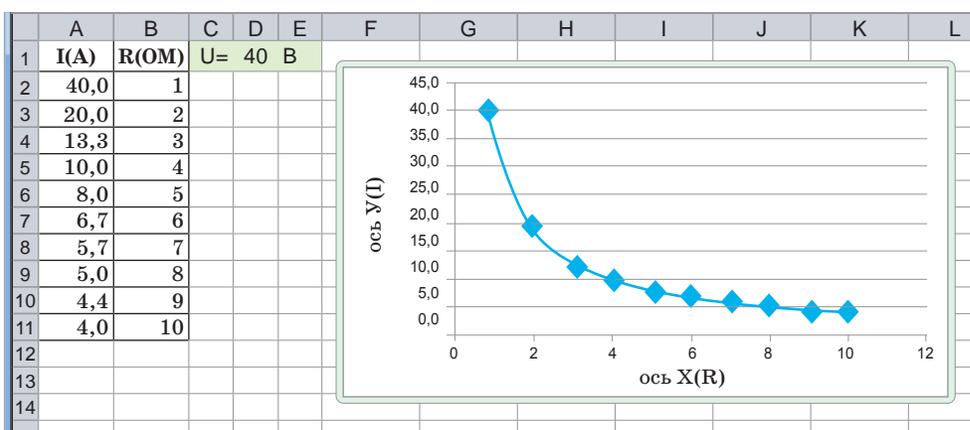


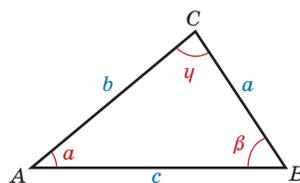
Рис. 2. График зависимости

Использование электронных таблиц для решения геометрических задач

Задание. Заданы все стороны нескольких треугольников. Используя формулу Герона, вычислите площади треугольников. Определите треугольник с наибольшей и наименьшей площадью.

Алгоритм выполнения задания

1. Используя заданные величины трех сторон (a , b и c) треугольника, вычислим его площадь (S) по формуле Герона из курса геометрии:



$$S = \sqrt{p(p-a)(p-b)(p-c)},$$

где p – полупериметр треугольника: $p = \frac{a+b+c}{2}$.

2. В приложении Excel на примере рисунка 3 заполним таблицу. Задайте свои значения сторон. По условию существования треугольника сумма любых двух его сторон должна быть больше третьей.
3. В ячейку E3 запишем формулу: $=\text{(B3+C3+D3)}/2$ и скопируем ее в ячейки с E4 по E12.
4. В ячейку F3 запишем формулу Герона: $=\text{КОРЕНЬ(E3*(E3-B3)*(E3-C3)*(E3-D3))}$ и скопируем ее в ячейки с F4 по F12.
5. Чтобы вычислить наибольшую площадь треугольника, используем формулу $=\text{МАКС(F3:F12)}$.
6. Чтобы вычислить наименьшую площадь треугольника, используем формулу $=\text{МИН(F3:F12)}$.

7. Чтобы вычислить среднее значение площади, используем формулу =СРЗНАЧ(F3:F12).

A15		fx Средняя площадь треугольника				
	A	B	C	D	E	F
1	Треугольник №	Стороны треугольника			Полупериметр	Вычисление площади треугольника
2		a	b	c		
3	1	10	12	13	17,5	57,00
4	2	8	7	9	12	26,83
5	3	5	4	3	6	6,00
6	4	8	5	11	12	18,33
7	5	4	6	8	9	11,62
8	6	7	11	13	15,5	38,50
9	7	14	15	16	22,5	96,56
10	8	2	4	3	4,5	2,90
11	9	4	4	7	7,5	6,78
12	10	3	4	5	6	6,00
13	Наибольшая площадь треугольника					96,56
14	Наименьшая площадь треугольника					2,90
15	Средняя площадь треугольника					27,05

Рис. 3. Вычисление в Excel

Применение электронных таблиц для решения экономических задач

Задание. Телефонная компания начала использовать новую систему оплаты за пользование телефоном. Если потребитель в течение одного месяца разговаривает до A минут, то за каждую минуту он платит 10 тенге, а если больше A минут, тогда за каждую минуту, превышающую норматив, оплачивает по B тенге. Как с помощью электронной таблицы вычислить сумму оплаты первых десяти потребителей?

Алгоритм выполнения задания

1. Для выполнения задания подготовим таблицу по образцу, представленному на *рисунке 4*.
2. Используя логическую функцию, вычислим стоимость разговоров, превышающих лимит A минут, а также стоимость разговоров, которые длились меньше A минут. Для этого в ячейку **C4** запишем формулу: =ЕСЛИ(B4>\$C\$1; \$C\$1*\$F\$1;B4*\$F\$1) и скопируем ее в ячейки с **C5** по **C13**. В формуле граница разговоров – A минут, ячейки со стоимостью разговоров имеют абсолютную адресацию.

- Если время разговоров потребителя превышает А минут, то лишние минуты занесем в столбец D. Для этого в ячейку D4 запишем формулу =ЕСЛИ (B4>\$C\$1;B4-\$C\$1;0) и скопируем ее в ячейки с D5 по D13.
- В ячейку E4 запишем формулу =D4*\$F\$2, вычисляющую стоимость разговоров, превышающих норматив, и скопируем ее в ячейки с E5 по E13.
- В столбце F будем вычислять стоимость разговоров за один месяц. Для этого в ячейку F4 запишем формулу =C4+E4 и скопируем ее в ячейки с F5 по F13.

	A	B	C	D	E	F
1	A=		до 300 минут	Цена = 10		
2	A=		больше 300 минут	Цена = 25		
3	Дома	Время, потраченное на разговоры по телефону за месяц (минуты)	Цена за разговоры (тенге)	Превышение допустимого времени (минут)	Цена за разговоры (тенге)	Общая сумма за телефонную связь (тенге)
4	Дом №1	500	3000	200	5000	8000
5	Дом №2	173	1730	0	0	1730
6	Дом №3	468	3000	168	4200	7200
7	Дом №4	392	3000	92	2300	5300
8	Дом №5	100	1000	0	0	1000
9	Дом №6	500	3000	200	5000	8000
10	Дом №7	267	2670	0	0	2670
11	Дом №8	308	3000	8	200	3200
12	Дом №9	150	1500	0	0	1500
13	Дом №10	87	870	0	0	870

Рис 4. Электронная таблица

Выполните задания, используя образцы решения прикладных задач.



Практическая работа

- Даны функции $y = 3x - 2$ и $y = -x + 4$. Постройте графики этих функций и найдите точку их пересечения.
- Постройте график функции $y = -x^2 + 2$.
- Вычислите НОД чисел 48, 84, 36, 60, 24.
- Вычислите НОК чисел 21, 32, 7, 6, 12.
- График зависимости между силой тока и напряжением $I(U)$ называют *вольт-амперной характеристикой*. Напряжение U изменяется от 1 до 10В с шагом 1В. Постройте график зависимости между силой тока и напряжением.
- Тело движется с постоянной скоростью 5 м/с. Постройте график зависимости пройденного пути от времени $S(t)$. Время изменяется в интервале $t = 0 \dots 20$ секунд с шагом 2 секунды.

7. Постройте график зависимости между кинетической энергией тела и его скоростью $E_k(V)$. Масса тела постоянна и равна $m = 100$ кг. Скорость тела изменяется в интервале $V = 0 \dots 20$ м/с с шагом 2 м/с. Определите, к какому виду относится график зависимости. Обоснуйте ответ.
8. Постройте график зависимости теплоемкости от температуры $Q(\Delta t)$. Какое количество тепла Q потребуется, чтобы изменить температуру воды массой $m=10$ кг. $C=4200$ Дж/кг $^\circ$ С. Интервал $\Delta t= 0 \dots 100^\circ$ С с шагом 10° С.
9. Используя данные, приведенные в таблице, вычислите неизвестные величины.

	А	В	С
1	Радиус окружности R (м)	Диаметр окружности d (см)	Длина окружности L (дм)
2	20	?	?
3	38	?	
4	15		
5	45		

10. В приведенной ниже таблице заданы координаты вершин треугольника. Например, одна вершина $A(x_1, y_1)$. Используя эти координаты, вычислите длины сторон треугольника. Определите, можно ли из этих сторон построить треугольник. Запишите «Да» или «Нет» рядом с величинами сторон в соответствующем столбце.

Примечание. Для нахождения длины сторон и определения условия существования треугольника воспользуемся формулой вычисления расстояния между двумя точками и условием существования треугольника.

	А	В	С	Д	Е	Ф	Г	Н	І	Ј
1	А		В		С		Длина стороны			Образует треугольник? «Да» или «Нет»
2	x_1	y_1	x_2	y_2	x_3	y_3	a	b	c	
3	2	3	6	8	17	8				
4	-4	-3	8	9	-7	1				
5	4	6	9	9	5	7				
6	9	8	90	99	3	3				
7	7	7	11	20	17	26				

11. Составьте таблицу, которая содержит баллы 10 учеников, сдававших ЕНТ по профильным предметам: математике и физике. Максимально можно набрать 45 баллов. Баллы переводятся в оценки в соответствующей шкале. Оценка «5» ставится за 81–100% правильных ответов, «4» – за 61–80%, «3» – за 41–60%. Если процент верных ответов ниже, то ставится оценка «2». Определите, сколько учеников по результатам тестирования получили оценки «5», «4», «3» и «2».

	А	В	С	Д	Е
1	Ученик	Математика	Оценка	Физика	Оценка
2	Ученик №1	25		18	
3	Ученик №2	40		29	
4	Ученик №3	12		14	
5	Ученик №4	32		36	
6	Ученик №5	40		40	
7	Ученик №6	20		5	
8	Ученик №7	13		6	
9	Ученик №8	7		21	
10	Ученик №9	4		18	
11	Ученик №10	35		25	
12	Количество оценок «2»				
13	Количество оценок «3»				
14	Количество оценок «4»				
15	Количество оценок «5»				

12. Даны демографические изменения численности населения Казахстана за последние десять лет. Составьте прогноз, какой будет численность населения страны к 2028 году.

№	А	В
1	Год	Численность населения
3	2011	16 559 000
4	2012	16 900 000
5	2013	17 035 000
6	2014	17 289 000
7	2015	17 557 000
8	2016	17 818 000
9	2017	18 014 000
10	2018	18 137 000
11	2019	18 632 500
12	2020	19 180 920
13	2021	?
14	2022	?

Творческие задания

Задание №1. Составьте таблицу, отражающую изменение атмосферного давления в течение недели в городе Нур-Султане. Проведите анализ изменения атмосферного давления. При выполнении задания используйте абсолютную адресацию и логические функции.

Нормальное атмосферное давление составляет 760 мм ртутного столба. Если значение атмосферного давления находится:

- в интервале 755–765 мм ртутного столба, то оно «соответствует норме»;

- в интервале 720–754 мм ртутного столба, то оно «ниже нормы»;
- в интервале 766–780 мм ртутного столба, то оно «выше нормы».

	А	В	С	Д	Е
1	№	Дни недели	Атмосферное давление (мм ртутного столба)	Отклонения от нормы	Оценка отклонения от нормы
2	1	Понедельник	746	16	Ниже нормы
3	2	Вторник	764	4	Соответствует норме
4	3	Среда	755	5	Соответствует норме
5	4	Четверг	778	32	Выше нормы
6	5	Пятница	719	41	Ниже нормы
7	6	Суббота	735	25	Ниже нормы
8	7	Воскресенье	758	2	Соответствует норме

Дескрипторы для выполнения задания:

- полностью форматирует таблицу;
- умеет использовать абсолютную адресацию ячеек;
- умеет правильно записывать в ячейки выражение с логическими функциями.

Задание №2. Определить физическое развитие учеников 8 класса.
 Норма веса = (Рост – Вес) · 1,1

	А	В	С	Д	Е	Ф
1	№	Ф. И. О. ученика	Рост (см)	Вес (кг)	Норма веса (кг)	Отклонение от нормы (%)
2	1	Ученик №1	161	49		
3	2	Ученик №2	157	50		
4	3	Ученик №3	149	55		
5	4	Ученик №4	172	60		
6	5	Ученик №5	165	50		
7		Среднее значение				

Дескрипторы для выполнения задания:

- полностью отформатировать таблицу;
- использовать формулы в ячейках;
- использовать типы данных в ячейках;
- использовать функции в ячейках.

Раздел III Обработка информации в электронных таблицах

3.8 Проектные работы в MS Excel



Как в повседневной жизни применять знания, полученные при изучении процессов обработки информации в электронных таблицах?



Подумай

- Как систематизировать теоретические знания и практические навыки, полученные при изучении возможностей электронных таблиц?

Темы для творческих работ

Выберите одну из тем. Выполните проект самостоятельно или в паре и презентуйте его. При выполнении задания, кроме рассмотрения теоретического материала, желательно создать программный продукт, чтобы получить высокие баллы.

Проект №1. Эффективное применение программы MS Excel в повседневной жизни

Краткое содержание

Раскрыть возможности программы MS Excel и показать, в каких сферах и как ее используют в повседневной жизни.

Проект №2. Диаграммы вокруг нас

Краткое содержание

Рассказать о видах диаграмм, которые можно построить на основе имеющихся данных в программе MS Excel. Показать историю использования диаграмм и способы их построения. Исследовать и доказать необходимость построения диаграмм на примерах из повседневной жизни.



электронная таблица – электрондық кесте – spreadsheet
график линейной функции – сызықтық функция графигі – graph linear functions

Проект №3. Исследование графиков линейной функции, параболы и гиперболы в программе MS Excel

Краткое содержание

Рассказать о видах математических функций, провести исследование графиков линейной функции, параболы и гиперболы на основе данных в программе MS Excel. Например, $y = ax^2 + b$.

Проект №4. Пути решения систем линейных уравнений в программе MS Excel

Краткое содержание

Провести исследование и составить алгоритм путей решения системы линейных уравнений в программе MS Excel. Исследовать пути представления, составления автоматизированных образцов, облегчающих вычисления в программе MS Excel. Представить программный продукт. Сделать презентацию.

Проект №5. Создание тестов в программе MS Excel

Краткое содержание

Исследовать возможности создания тестов в приложении MS Excel. Объяснить особенности составленного теста по сравнению с аналогами. Выявить достоинства и недостатки, представить свой программный продукт. Создать презентацию.

Проект №6. Эффективное использование программы MS Excel для определения качества знаний

Краткое содержание

Объяснить, как с помощью программы MS Excel можно оценить качество знаний учащегося в каждой четверти. Составить автоматизированные образцы для расчета качества знаний и представить программный продукт. Создать презентацию.

Проект №7. Увеличение точности вычислений в лабораторных работах по физике с помощью программы MS Excel

Краткое содержание

В курсе школьной физики проводить вычисления в лабораторных работах вручную нелегко. Представить образцы программ для точного вычисления абсолютной и относительной погрешностей при проведении расчетов в программе MS Excel. Представить программу. Создать презентацию.

Проект №8. Использование программы MS Excel для решения прикладных математических задач.

Краткое содержание

Представить исследование использования программы MS Excel для решения прикладных математических задач, а также о применении математики в повседневной жизни. Разработать программу. Сделать презентацию.

Рекомендации по эффективной защите проекта

- Назовите тему проекта.
- Объясните выбор темы.
- Сформулируйте цели и задачи.
- Опишите, какие использовали средства.
- Сделайте краткий обзор темы.
- Поделитесь мыслями о реализации проекта.
- Расскажите о новых знаниях и навыках, полученных при работе над проектом.
- Познакомьте слушателей с заключением проекта.
- Выслушайте мнения и оценки аудитории, сделайте выводы.

Критерии успешного проекта

- Самостоятельная реализация.
- Актуальность темы.
- Раскрытие темы.
- Свой способ представления решения проблемы.
- Ораторское мастерство при защите проекта и умение держать себя на публике.
- Умение использовать технические и визуальные средства.
- Умение правильно и точно отвечать на поставленные вопросы.

Отчет о выполненном проекте должен содержать ответы на вопросы

- Почему я выбрал эту тему?
- Что нового я узнал, чему научился?
- Как я осуществлял поиск необходимой информации?
- Как я редактировал информацию?
- Как я совмещал теорию и практику?
- Какие части проекта наиболее эффективны?
- Что я не смог выполнить?
- Что было самым легким в проекте?
- Какие трудности возникали во время работы над проектом?
- Что бы я изменил в проекте, если бы начал его заново?
- Мое впечатление от работы над проектом.



Как можно использовать оператор цикла с параметром на языке программирования Python?



Подумай

- Что вы знаете о циклах?
- Какова роль циклов в природе и жизни человека?
- Что относится к циклам в математике?



Новые знания

Природа и жизнь человека состоят из определенных циклов. Можно сказать, что цикл представляет собой совокупность постоянно повторяющихся событий. В качестве примера повторяющихся в природе циклов можно привести смену времен года и дней недели. Видов циклов много. К ним относятся экономические, исторические, программные, математические, жизненные циклы и др. В программировании большое значение имеют циклические алгоритмы. С их помощью можно составлять алгоритм всех видов циклов, встречающихся в природе, и моделировать их на компьютере.

Алгоритмы, характеризующиеся многократным выполнением определенных действий, называют **циклическими алгоритмами**.

Циклические алгоритмы можно кратко записать с помощью оператора языка программирования. Существуют специальные операторы, которые переводят циклические алгоритмы на язык программирования. В зависимости от вида применения циклические операторы классифицируются на два вида (*схема 1*).

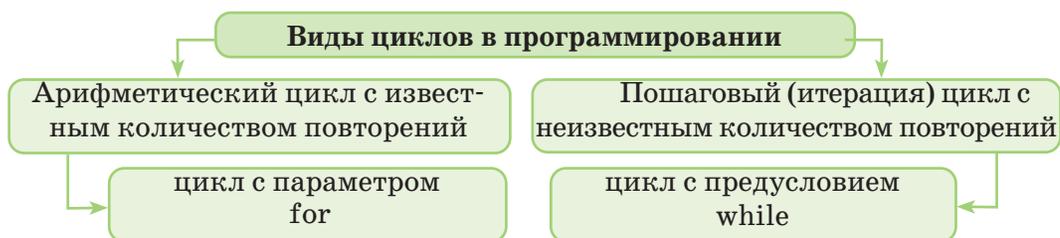


Схема 1. Виды циклов в программировании

Если при разработке алгоритма необходимо организовать заранее определенный процесс, то используется оператор цикла с параметром. К такому оператору относится оператор **for**. В Python оператор **for** выполняет вычисления до достижения параметром значения, указанного заранее. Для организации работы цикла с параметром необходимо использовать оператор **for** вместе с функцией **range()**.

Запись:

```
for i in range(k, n, x):
    <оператор или выражение>;
```

функция **range(k, n, x)** создает последовательность чисел с шагом **x** от **k** до **n - 1**.

Например, если **k = 1**, **n = 11** и **x = 2**, то организацию цикла **for i in range(1, 11, 2)** следует читать: значение величины **i** изменяется от 1 до 10 с шагом 2.

Способы использования функции **range()** оператора **for**

for i in range(n):	for i in range(k, n):	for i in range(k, n, x):
<pre>for i in range(6): print(i)</pre>	<pre>for i in range(6, 11): print(i)</pre>	<pre>for i in range(1, 11, 2): print(i)</pre>
Результат:	Результат:	Результат:
0 1 2 3 4 5	6 7 8 9 10	1 3 5 7 9

Существуют два вида использования оператора **for**:

- 1) значение параметра цикла увеличивается (Пример 1);
- 2) значение параметра цикла уменьшается (Пример 2).

Рассмотрим простые примеры, чтобы понять работу цикла.

Пример 1. Разработайте программу, вычисляющую сумму чисел от 1 до 100. Код программы примера приведен на *рисунке 1*. При выполнении кода программы на экран выходит число 5050.

```
4.1.1.py - C:\Users\User\AppData...
File Edit Format Run Options Window Help
i=s=0
for i in range (1,101):
    s+=i
print(s)
Ln:3 Col:8
```

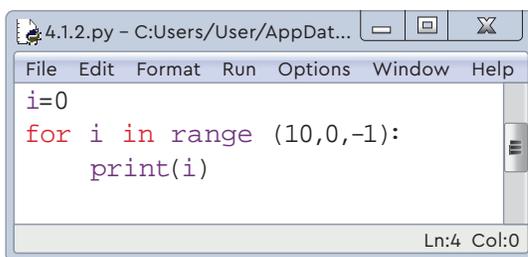


Рис. 1. Программный код примера 1



цикл - цикл - cycle
диапазон - ауқымы - range

Пример 2. Выведите на экран числа от 1 до 10 в обратном порядке. Код программы приведен на *рисунке 2*. На экран при выполнении кода программы выводится последовательность чисел 10 9 8 7 6 5 4 3 2 1.



```
4.1.2.py - C:\Users\User\AppData...
File Edit Format Run Options Window Help
i=0
for i in range (10,0,-1):
    print(i)
Ln:4 Col:0
```

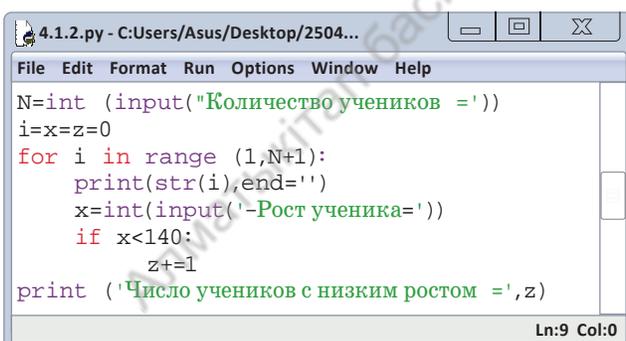
Рис. 2. Программный код примера 2



Практическая работа

В 8 «А» классе учатся N учеников ($0 < N < 30$). Дан рост учащихся. Если рост учащегося меньше 140 см, то он относится к ученикам с низким ростом. Напишите программный код, определяющий, сколько учеников в классе низкого роста.

Дадим пояснение коду программы по выполнению практической работы (*рис. 3*). При разработке программы по условию за-



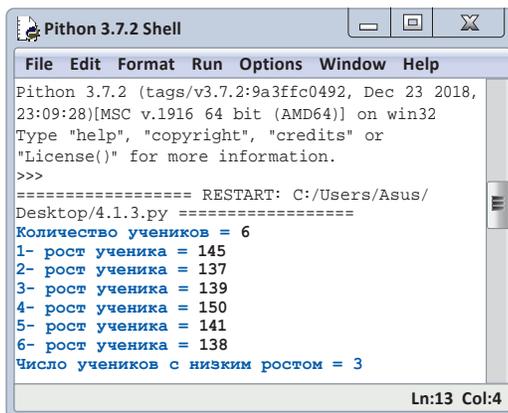
```
4.1.2.py - C:\Users\Asus\Desktop\2504...
File Edit Format Run Options Window Help
N=int (input("Количество учеников ="))
i=x=z=0
for i in range (1,N+1):
    print(str(i),end='')
    x=int(input('-Рост ученика='))
    if x<140:
        z+=1
print ('Число учеников с низким ростом =',z)
Ln:9 Col:0
```

Рис. 3. Программный код

дачи будем использовать совместно циклические алгоритмы и алгоритмы разветвления. Чтобы ввести N раз рост всех учащихся, необходимо записать оператор ввода внутри цикла и считывать рост учащегося из переменной x на каждом шаге цикла. На следующем шаге мы сравниваем рост каждого ученика со 140. Как видно, в теле цикла выполняется не один, а три оператора. Для использования трех операторов в одном цикле после оператора **for** необходимо поставить 4 пробела. В цикле $z + = 1$ подразумевает краткую запись выражения $z = z + 1$, то есть увеличивает значение z на 1. Строка **print (str (i), end="")** в цикле используется, чтобы указать порядковый номер ученика. Результаты работы программы приведены на *рисунке 4*.

Бесконечный цикл

В некоторых программах управление циклом становится невозможным. Разумеется, это случай, вызванный ошибками, допущенными при программировании. Такой цикл называют **бесконечным циклом**. Чтобы остановить программу в процессе бесконечного цикла, необходимо одновременно нажать на комбинацию клавиш **Ctrl + C**.



```
Python 3.7.2 Shell
File Edit Format Run Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018,
23:09:28)[MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or
"License()" for more information.
>>>
===== RESTART: C:/Users/Asus/
Desktop/4.1.3.py =====
Количество учеников = 6
1- рост ученика = 145
2- рост ученика = 137
3- рост ученика = 139
4- рост ученика = 150
5- рост ученика = 141
6- рост ученика = 138
Число учеников с низким ростом = 3
Ln:13 Col:4
```

Рис. 4. Результат выполнения программы



Анализ



Проанализируйте код программы, разработанный в практической работе (рис. 3). Как нужно перефразировать условие задачи и какие изменения необходимо сделать в коде программы для определения количества учащихся с ростом от 140 до 155 см? Обоснуйте свой ответ.



Синтез



Составьте условие одной простой задачи, решаемой с помощью цикла с параметром. Разработайте и представьте проект кода программы на языке Python, составленный для данной задачи.



Оценка



Оцените значимость использования циклических операторов в программировании.



Вопросы для размышления

1. Как вы понимаете термин «цикл»?
2. Как циклы влияют на жизнь человека?
3. Какие виды циклов существуют в языке программирования Python? В чем их отличие?
4. Чем циклический алгоритм отличается от разветвляющегося?

**Задание**

*

Задание №1. Разработайте программу, определяющую все четные числа, расположенные между K и N ($1 \leq K \leq N \leq 100$), и выведите их на экран.

Например	Результат
23 35	24 26 28 30 32 34

*

Задание №2. Среди всех двузначных чисел, расположенных до заданного N ($10 \leq N \leq 99$), найдите числа с одинаковыми цифрами в десятках и единицах.

Например	Результат
34	11 22 33

**

Задание №3. Идет отбор учеников 8-го класса для записи в баскетбольную секцию с учетом их роста. Всего в отборе приняли участие N ($130 \leq \text{рост учащихся} \leq 200$) учащихся. Из них были приняты в секцию ребята с ростом выше K сантиметров. Сколько учащихся принято в секцию и какой рост у самого высокого ученика? Рост учащихся вводится последовательно.

Например	Результат	Пояснение
7 170 165 180 171 135 143 175 169	3 2	Выше 170 см есть 3 ученика. Самый высокий ученик на 2-м месте.

**Домашнее задание**

Когда программа завершает работу, какие значения выводит на экран оператор `print()`, отображаемый в программе?

1

```
N = 5
i = x = z = 0
for i in range(1, N + 1):
    x = int(input())
    if x % 10 == 4:
        z += 1
print(z)
```

2

```
N = 5
i = x = s = 0
for i in range(1, N + 1):
    x = int(input())
    if x % 2 == 1:
        print(x % 10)
        s += x % 10
print(s)
```



Как организовать встроены́е циклы на языке программирования Python?



Подумай

- Что такое вложенный цикл?
- Какой результат можно получить, если использовать два цикла for, вложенных друг в друга?
- Для решения каких задач необходимо использовать вложенные циклы?



Новые знания

Несколько простых циклов, один из которых используется внутри другого, называются **вложенными циклами**. Во вложенных циклах внутренний цикл должен полностью находиться внутри внешнего цикла. Вид записи двух простых циклов, вложенных друг в друга, приведен на *схеме 1*. Вложенный цикл также называют **сложным циклом**.

Число n , используемое в цикле, относится к типу целых чисел (*схема 1*). Если в программе используется один цикл, то он работает $n-1$ раз. Если в программе используются два вложенных цикла (пример 1), то эта группа циклов работает $n \cdot n$ раз. Например, внешний цикл, приведенный ниже, работает 10 раз, а внутренний – 100 раз.

```

for i in range(1, n):
    for j in range(1, n):
        <1-оператор>
        <2-оператор>
        ...

```

Схема 1. Вид записи двух простых циклов

1 for i in range(1, 11):
for j in range(1, 11):

2 for i in range(1,101):
for j in range(1, 101):
for k in range(1, 101):

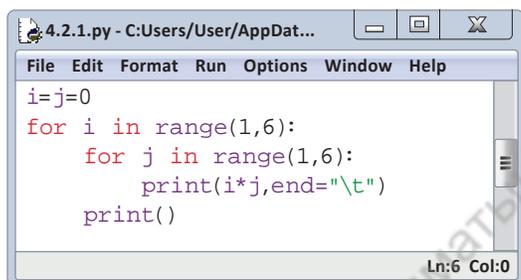
Вложенные циклы целесообразно использовать для решения задач, в которых рассматривается множество вариантов. Но использование более 2–3 вложенных друг в друга циклов неэффективно. Это



вложенный цикл – кірістірілген цикл – embedded loop
выполнение цикла – циклды орындау – loop execution

продлевает время работы программы. Если в программе используются три простых цикла (пример 2), то время работы программы равно n^3 . По следующему примеру можно понять, что эта величина очень большая. Например, если $n = 100$, то три вложенных цикла будут выполняться 1010100 раз. Если учесть, что цикл будет выполняться около 1 000 000 раз, то при $n = 1000$ цикл будет работать 10^9 раз, то есть миллиард раз. Это займет около 1000 секунд, то есть примерно 17 минут. Чтобы дождаться результатов одного вычисления на компьютере, необходимо примерно 17 минут. Для этого при решении таких задач предусматривается уменьшение работы цикла с помощью эффективных алгоритмов.

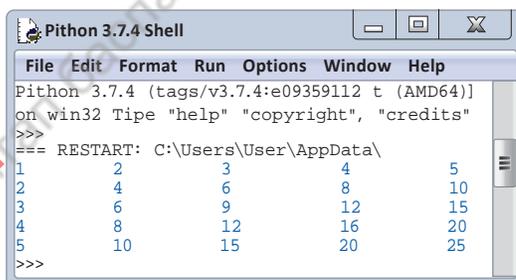
С помощью вложенных циклов можно выводить числовые и символьные данные на экран в виде прямоугольной таблицы. Числовые данные удобны для понимания в виде прямоугольной таблицы. Для вывода данных на экран в виде прямоугольной таблицы нужно записать определенную последовательность операторов. Например, для построения таблицы умножения размером 5x5 в виде прямоугольника можно записать код программы, представленный на *рисунке 1*, а результат выполнения программы приведен на *рисунке 2*.



```
4.2.1.py - C:\Users\User\AppData\Local\Microsoft\Windows\Apps\PythonSoftwareFoundation.Python.3.7_...
File Edit Format Run Options Window Help
i=j=0
for i in range(1,6):
    for j in range(1,6):
        print(i*j,end="\t")
    print()
Ln:6 Col:0
```



Рис. 1. Код программы



```
Python 3.7.4 Shell
File Edit Format Run Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112 t (AMD64))
on win32 Type "help" "copyright", "credits"
>>>
=== RESTART: C:\Users\User\AppData\Local\Microsoft\Windows\Apps\PythonSoftwareFoundation.Python.3.7_...
1      2      3      4      5
2      4      6      8     10
3      6      9     12     15
4      8     12     16     20
5     10     15     20     25
>>>
```

Рис. 2. Результат программы

Как видно из кода программы, пока внешний цикл выполняется один раз, внутренний цикл выполняется пять раз. При выводе данных в прямоугольной форме важно правильно разместить оператора `print()`. Здесь `"\t"` позволяет установить одинаковое расстояние между данными через табулирование. При использовании вложенных циклов часто встречаются ошибки.

Распространенные ошибки в использовании вложенных циклов

- запись одинаковых параметров цикла для внутренних и внешних циклов;
- в зависимости от ошибочного количества пробелов при размещении операторов во внутреннем-внешнем циклах вложенных циклов возможны случаи, когда оператор или выражение не вошли в цикл;
- использование действительных чисел в качестве параметра цикла.



Практическая работа

Задание. Простое число. Дана последовательность чисел длиной N ($N < 100$). Определите простые числа в заданной последовательности чисел и выведите их на экран. Простое число – это число, которое делится только на себя и на один.

Алгоритм решения задачи. Для решения поставленной задачи необходимо использовать вложенные циклы. Чтобы проверить, является ли число простым, квадратный корень из этого числа округляем до целого значения. Мы используем внешний цикл для ввода чисел, которые нам нужно проверить. Внутренний цикл используется для проверки заданных чисел на выполнение условия простого числа. Параметр внутреннего цикла начинается с 2 и изменяется одним шагом до корня числа, округленного до целого значения. Если среди этих чисел нет делителя проверяемого числа, оно относится к простому числу. На *рисунке 2* представлен код программы, а результат выполнения программы показан на *рисунке 3*.

Например	Результат
5	5 197 47
5	
45	
197	
47	
18	

```

import math
N=int(input())
i=j=x=0
for i in range(1,N+1):
    x=int(input())
    z=0
    for j in range(2,round(math.sqrt(x))):
        if x%j==0:
            z=1
    if z==0:
        print(x)
  
```

Рис. 2. Код программы определения простых чисел

```

Python 3.7.4 Shell
Python 3.7.4 (tags/v3.7.4:e (AMD)) on win32
Type "help" "copyright", "credits"
>>> RESTART: C:\Users\User\AppData\
5
5
45
197
47
18
5
197
47
>>>
  
```

Рис. 3. Результат программы

В Python для использования математических функций используются специальные модули. Чтобы запустить математический модуль в программе, используем команду `import`. Например, `math.sqrt(x)` дает квадратный корень из числа x , тогда `round(math.sqrt(18))=4`.



Анализ



Проанализируйте код программы «Определение простых чисел», рассмотренный в практическом задании. Запишите пояснение каждой строки программы.

**Вопросы для размышления**

1. В чем отличие вложенных и простых циклов?
2. Как можно высчитать время выполнения вложенных циклов?
3. Каковы негативные последствия использования нескольких циклов на работу программы?
4. Функции какого оператора при написании программы вывода данных в виде таблицы являются важными?

**Задание**

Задание №1. Организуйте вывод данных в прямоугольном виде для $n = 5$, используя вложенные циклы. Рекомендации для выполнения задания: лучше использовать циклы вместе с оператором проверки условия.

1)	2)	3)	4)	5)
1 1 1 1 1	1 1 1 1 1	1 1 1 1 1	1 0 0 0 0	0 0 0 0 1
1 1 1 1 1	0 0 0 0 0	2 2 2 2 2	0 1 0 0 0	0 0 0 1 0
1 1 0 1 1	1 1 1 1 1	3 3 3 3 3	0 0 1 0 0	0 0 1 0 0
1 1 1 1 1	0 0 0 0 0	4 4 4 4 4	0 0 0 1 0	0 1 0 0 0
1 1 1 1 1	1 1 1 1 1	5 5 5 5 5	0 0 0 0 1	1 0 0 0 0

Задание №2. Какое значение будет присвоено переменной x после завершения программы?

1

```
i = j = x = 0
for i in range(1, 6):
    for j in range(i + 1, 6):
        x += 1
```

2

```
i = j = x = 0
for i in range(1, 6):
    for j in range(i + 1, 6 - i):
        x += 1
```

3

```
i = k = x = 0
for i in range(1, 5):
    for k in range(i - 1, i + 1):
        x += 2
```

4

```
i = k = 0; x = 10
for i in range(1, 4):
    for k in range(0, i):
        x = x + (k - 1)
```

**Домашнее задание**

Составьте условие одной задачи на тему «Вложенные циклы». Предложите алгоритм решения задачи и разработайте соответствующую программу.



Как используется оператор условного цикла в языке программирования Python?



Подумай

- Что такое циклический процесс?
- Какие примеры из повседневной жизни можно отнести к циклическим процессам?
- Какие виды циклов существуют в программировании?
- Как вы могли бы связать понятия «условие» и «цикл»? Приведите пример.



Новые знания

Изучая работу цикла, вы познакомились с параметром **for**. Теперь давайте рассмотрим работу **while**, который является вторым видом цикла. Порядок работы цикла условий **while** приведен на схеме 1. Цикл условия содержит зарезервированное слово **while** (пока), и условие цикла состоит из выражения логического типа **<условие>**. Условие цикла пишется в виде логического выражения. Результатом логического выражения является результат булевого типа.

While <условие>:

```

тело цикла
<оператор 1 или выражение>
<оператор 2 или выражение>
...
<оператор n или выражение>

```

Схема 1. Вид записи цикла **while**

Логическое выражение цикла условия принимает значение только **true** (истина) или **false** (ложь). Перед началом выполнения операторов внутри цикла проверяется значение условия цикла. Если условие принимает истинное значение, то цикл продолжает свою работу. Если значение условия является ложным, то цикл завершает работу. Для



цикл **while** («пока») – **while** («эзірге») циклы – **while loop**
условие цикла – цикл шарты – **loop condition**
блок-схема – блок-схема – **flowchart**

того чтобы разместить другой оператор или выражение в теле цикла, необходимо поставить 4 пробела в строках после строки, содержащей оператор **while** и условие цикла. Операторы и выражения внутри цикла выполняются последовательно, пока условие цикла не примет ложное значение. Если условие ложное, то тело цикла перестает работать, не начиная выполнения.

Рассмотрим примеры для понимания основ работы оператора **while**. Для этого снова разберем пример 1, приведенный в теме «Циклы с параметром». Выбор этого примера обоснован тем, что знакомство с работой двух циклов на одном примере позволяет облегчить понимание особенностей их работы.

Пример: Разработайте программу, которая вычислит сумму всех чисел от 1 до 100.

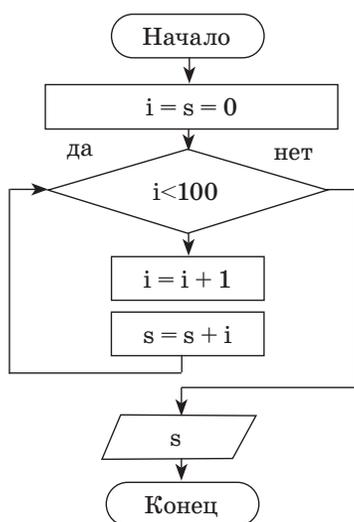


Рис. 1. Блок-схема

В данном примере сложение чисел от 1 до 100 можно выполнить с использованием цикла условия (рис. 1). Вопрос условия проверки «Значение величины *i* цикла меньше 100?» повторяется каждый раз. Блок-схема циклического алгоритма представлена на рисунке 1. Здесь работа цикла остановится, когда значение величины *i* превысит 100. До тех пор, пока условие цикла не является ложным, к значению суммы *s* прибавляется значение *i* (рис. 2).

```

432.py - C:Users...
File Edit Format Run Options Window
Help
i=s=0
while i<100:
    i+=1
    s+=i
print(s)
Ln:6 Col:0
    
```

Рис. 2. Программный код задачи

Особенности цикла while:

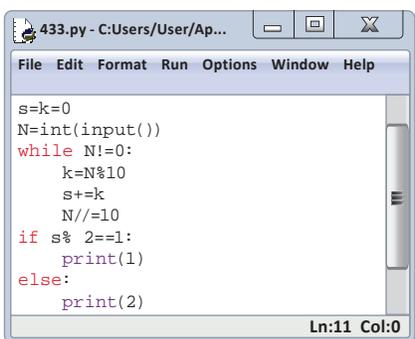
- цикл позволяет выполнять одну и ту же последовательность действий, пока проверяемое условие истинно;
- цикл условия применяется при решении задач, когда число циклов неизвестно.



Практическая работа

Задача. Определите, четной или нечетной является сумма цифр заданного натурального числа *N*. Если сумма является нечетным числом, то выведите на экран «1», иначе – «2».

Например	Результат
4856	1
7517	2



```

433.py - C:\Users\User\Ap...
File Edit Format Run Options Window Help
s=k=0
N=int(input())
while N!=0:
    k=N%10
    s+=k
    N//=10
if s% 2==1:
    print(1)
else:
    print(2)
Ln:11 Col:0

```



Рис. 3. Программный код задачи

Чтобы разработать код программы задачи, возникает только одна проблема. Как можно отделить цифры числа, чтобы их использовать? Для рассмотрения цифр числа N в отдельности обратимся к закономерностям в математике. Одна из таких закономерностей: при делении любого числа на 10 остаток равен последней цифре числа. Если $N = 148$, то $148 : 10 = 14$ и остаток равен 8. Последняя цифра числа равна 8. Мы используем эту закономерность в качестве алгоритмического решения поставленной задачи при разработке программы (рис. 3).



Анализ



Выполните анализ кода программы задачи, разработанной в практической работе (рис. 3). Опишите функции, выполняемые каждой строкой программы.



Синтез



В практической работе необходимо изменить условия заданной задачи так, чтобы нужно было определить нули в составе заданного числа (рис. 3). Какие изменения необходимо сделать в коде программы? Предложите изменить код программы. Обоснуйте ваши предложения.



Оценка



Обмен мнениями. Сравните основные сходства и различия в деятельности цикла условий **while** и цикла с параметром **for**. Можно ли оставить один из двух операторов и не использовать другой? Если необходимо сократить количество операторов цикла, то какой нужно оставить?

**Вопросы для размышления**

1. Что такое цикл условия?
2. Каковы основы работы оператора **while**?
3. В каком случае останавливается цикл условия?
4. Какие задачи, использующие цикл условия, вы можете привести в качестве примеров?

**Задание**

*

Задача №1. Дано натуральное число N . Составьте блок-схему, которая находит произведение цифр этого числа. Напишите программный код задачи.

Например	Результат
4582	320

*

Задача №2. Дано натуральное число N . Составьте блок-схему, чтобы определить, сколько цифр в этом числе. Напишите код программы.

Например	Результат
8361270	7

**

Задача №3. Дано натуральное число N . Создайте блок-схему, которая находит наибольшую из цифр этого числа. Напишите программный код задачи.

Например	Результат
70856954	9

**

Задача №4. Проанализируйте заданный программный код. Каким будет результат на экране, когда программа выполнится? Посмотрите на результаты программы и определите условие задачи.

1

```
x = 0
x = int(input() )
while x != 0:
    k = x%10
    print(k, end="")
    x //= 10
```

2

```
x = 0
x = int(input() )
while x != 0:
    k = x%2
    print(k, end="")
    x //= 2
```

**Домашнее задание**

Дано натуральное число N . Сколько нечетных цифр в этом числе? Составьте блок-схему и напишите программный код.

Например	Результат
3857455	5



Как эффективно использовать инструкцию управления циклом `continue` в программировании на языке Python?



Подумай

- Может ли человек вмешаться или управлять определенным циклическим процессом, который происходит в повседневной жизни? Приведите примеры.
- Что можно сделать, чтобы перейти к следующему шагу, упустив некоторые предыдущие шаги цикла при выполнении программы?



Новые знания

В процессе выполнения программы существует много условий, при которых приходится управлять работой цикла. При программировании шаг текущего цикла (итерация) может остановить работу по заданному условию и сразу перейти на следующий шаг цикла. Это может сделать инструкция `continue`.

Continue (продолжение, продолжить) – инструкция цикла, которая с помощью определенного условия остановит итерацию текущего цикла и переведет на итерацию следующего цикла.

Инструкция `continue` обеспечивает мгновенный переход к началу цикла. С помощью этого руководства в некоторых случаях можно предотвратить использование циклов, вставляя их. Инструкцию `continue` можно использовать в двух видах цикла: в цикле с параметром `for` и в цикле условия `while` (схемы 1 и 2).

for i in range (k, n, x):

```
<оператор 1 или выражение>
<оператор 2 или выражение>
if <условие>:
    continue
...
< оператор n или выражение >
```

Схема 1. Использование в цикле `for`

while <условие>:

```
<оператор 1 или выражение>
<оператор 2 или выражение>
if <условие>:
    continue
...
< оператор n или выражение >
```

Схема 2. Использование в цикле `while`



управление циклом – циклды басқару – cycle control
переход к началу цикла – циклдің басына бару – go to the beginning of the cycle

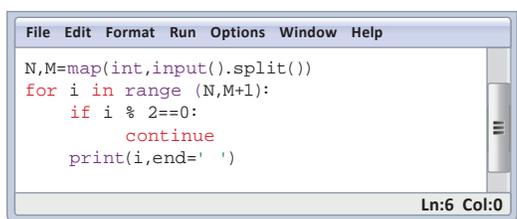
Рассмотрим примеры использования инструкции **Continue**, чтобы понять, как она работает в двух видах цикла.

Пример 1. Составьте код программы, который выводит на экран все нечетные числа между N и M целыми числами.

По условию примера организуем цикл между N и M и проверим четность-нечетность значения параметра цикла. Если параметр цикла является четным, то происходит возврат к началу цикла и выполняется последующий шаг цикла. Код программы представлен на *рисунке 1*, а результат показан на *рисунке 2*.

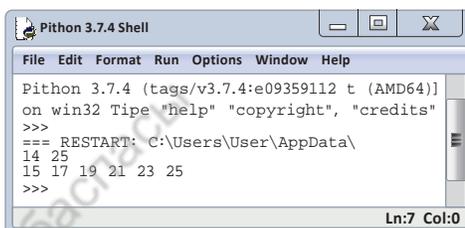
Например	Результат
14 25	15 17 19 21 23 25

Использование инструкции `continue` в цикле с условием `while`



```
File Edit Format Run Options Window Help
N,M=map(int,input().split())
for i in range (N,M+1):
    if i % 2==0:
        continue
    print(i,end=' ')
Ln:6 Col:0
```

Рис. 1. Программный код (for)

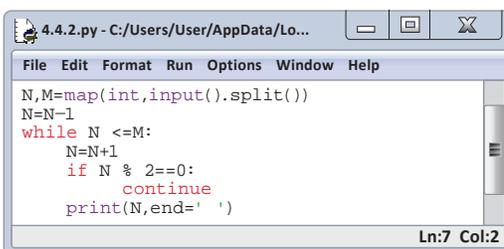


```
Python 3.7.4 Shell
Python 3.7.4 (tags/v3.7.4:e09359112 t (AMD64))
on win32 Tipe "help" "copyright", "credits"
>>>
=== RESTART: C:\Users\User\AppData\
14 25
15 17 19 21 23 25
>>>
```

Рис. 2. Результат программы

Использование инструкции `continue` в цикле с параметром `for`

На *рисунке 3* представлена программа использования инструкции `continue` в цикле `while`. В использовании обоих циклов нет большого различия. При выполнении кода программы получаем результат, представленный на *рисунке 2*. Если есть необходимость решения задач такого типа, то можно использовать любой из двух циклов.



```
4.4.2.py - C:/Users/User/AppData/Lo...
File Edit Format Run Options Window Help
N,M=map(int,input().split())
N=N-1
while N <=M:
    N=N+1
    if N % 2==0:
        continue
    print(N,end=' ')
Ln:7 Col:2
```

Рис. 3. Программный код (while)

Эффективное использование инструкции **continue** помогает упростить структуру цикла в программе. С помощью этой инструкции удобно упрощать некоторые задачи в программе. Это можно увидеть на примере ниже.

Пример 2. Дана строка S , состоящая из латинских букв. В этой строке удалите все буквы « a » и выведите оставшуюся строку на экран.

Для данного примера можно легко создать программу, используя цикл с параметром

for с помощью инструкции **continue**. На *рисунке 4* представлен код программы задачи, а результат показан на *рисунке 5*.

Например	Результат
Aduyaaaakeraasd	duykersd

```

4.4.3.py - C:/Users/User/AppData/Lo...
File Edit Format Run Options Window Help
s=input('s=')
for i in S:
    if i=='A' or i=='a':
        continue
    print(i,end=' ')
Ln:5 Col:15
  
```

Рис. 4. Программный код (пример 2)

```

Python 3.7.4 Shell
File Edit Format Run Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112 t (AM
on win32 Tipe "help" "copyright", "credi
>>>
=== RESTART: C:\Users\User\AppData\
S=aduyaaaakeraasd
duykersd
>>>
  
```

Рис. 5. Результат программы

Работа с циклом **for** была более сложной, чем с циклом **while**, но в Python этот цикл наиболее универсален. Его можно использовать для любого из объектов, таких как последовательность чисел, строки и списки. В Примере 2 мы познакомились с методом использования цикла **for** к строковым объектам.



Вопросы для размышления

1. Какова функция инструкции **continue**?
2. Какие ошибки возможны при использовании инструкции **continue** в создании программы?
3. Приведите примеры использования инструкции **continue**.



Анализ

Сделайте анализ кода программы примера 2 (*рис. 4*). Можно ли реализовать этот пример с использованием цикла **while** вместо цикла **for**? Если да, то какие изменения необходимо внести в код программы. Объясните причины.



Синтез

Выберите одну из известных вам циклических задач. Предложите способ решения с помощью инструкции **continue**.

**Задание**

*

Задача №1. Напишите программный код, который отображает на экране все числа между K и P , у которых последняя цифра равна t .

Например	Результат
10 39 7	17 27 37

**

Задача №2. Дано положительное целое число N . Разработайте программный код, который в этом числе удалит все цифры, равные P , и отобразит оставшиеся цифры в обратном порядке.

Например	Результат
2547585 5	8742

**

Задача №3. Дана строка S , состоящая из строчных букв английского алфавита. Создайте программный код, который из этой строки удаляет гласные и отображает на экране только согласные. В английском языке есть шесть гласных букв: а, е, і, о, и у.

Например	Результат
knkauseqiytraa	knksqtr

**

Задача №4. В марте во всех регионах Казахстана погода переменчивая. С наступлением весны солнце прогревает землю, но в некоторые дни температура опускается ниже нуля. Приводится средняя температура воздуха в стране в марте между A и B ($1 \leq A \leq B \leq 31$). Создайте программу с использованием `continue`, которая будет определять количество дней, когда температура выше нуля. Ввод данных осуществляется последовательно.

Например	Результат
10 19 -5 3 4 -2 3 -1 4 7 10 7	7

**Домашнее задание**

Даны целые числа N и M . Напишите программный код, который суммирует каждое третье число, начиная с N .

Например	Результат	Объяснение
21 34	135	$21 + 24 + 27 + 30 + 33 = 135$



В каких случаях можно использовать инструкцию по управлению циклом `break` в программировании на языке Python?



Подумай

- Можно ли при необходимости остановить определенный циклический процесс, который происходит в повседневной жизни? Приведите примеры.
- Что делать, чтобы приостановить работу цикла после достижения соответствующего результата при выполнении программы?



Новые знания

В ходе разработки программы при изучении инструкции `continue` вы узнали, что можно перейти на следующий шаг, не выполнив некоторые шаги цикла. Во время программирования работа цикла может быть прекращена, когда получен желаемый результат. Для этого используется инструкция `break`.

Break (прерывать, ломать) – инструкция цикла, которая немедленно прекращает работу текущего цикла после выполнения определенных условий, выводит работу программы из цикла и позволяет продолжить ее.

Инструкция `break` в Python позволяет вам остановить циклический процесс при возникновении внешнего фактора, влияющего на работу цикла. Инструкцию `break` можно использовать в цикле с параметром `for` и в цикле с условием `while` (схемы 1 и 2). Как видно из схем, запись `break` в программе не отличается от записи инструкции

```
for i in range (k,n,x):
    <оператор 1 или выражение >
    <оператор 2 или выражение >
    if <условие>:
        break
    ...
    <оператор n или выражение>
```

Схема 1. Использование цикла `break` в цикле `for`

```
while <условие>:
    <оператор 1 или выражение >
    <оператор 2 или выражение >
    if <условие>:
        break
    ...
    <оператор n или выражение>
```

Схема 2. Использование цикла `break` в цикле `while`



прерывание цикла – циклдың үзілуі – conditional break
инструкция цикла – цикл нұсқаулығы – loop instruction

continue. Только они выполняют две различные функции: одна обеспечивает выход из цикла, другая позволяет пропустить выполнение определенных шагов цикла.

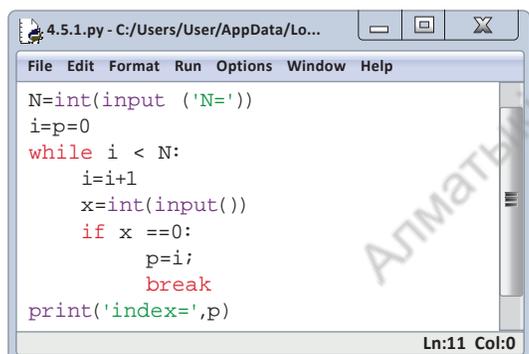
Рассмотрим пример, чтобы понять, как инструкция **break** работает в двух видах цикла.

Пример. Дана последовательность чисел длиной N ($1 \leq N \leq 100$). Среди чисел последовательности только одно равно нулю. Напишите программу, которая определит порядковый номер расположения нуля.

Программа данного примера создана с помощью **while** и **break** (рис. 1). При выполнении кода программы получаем результат, представленный на рисунке 2. При вводе нуля в последовательность чисел программа сохраняет его порядковый номер и останавливает работу цикла через **break** (рис. 2).

Например	Результат
N=5 4 8 0 7 4	3

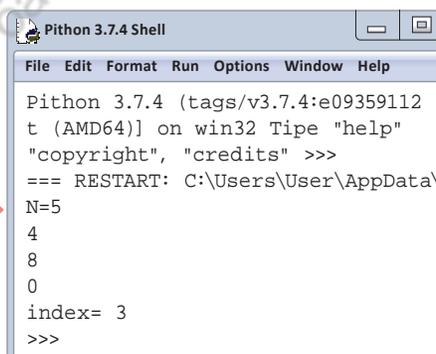
Использование инструкции break в цикле условия while



```

4.5.1.py - C:/Users/User/AppData/Lo...
File Edit Format Run Options Window Help
N=int(input ('N='))
i=p=0
while i < N:
    i=i+1
    x=int(input())
    if x ==0:
        p=i;
        break
print('index=',p)
Ln:11 Col:0
  
```

 Рис. 1. Код программы (while)



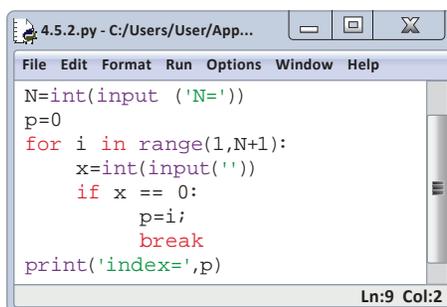
```

Python 3.7.4 Shell
File Edit Format Run Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112
t (AMD64)] on win32 Tipe "help"
"copyright", "credits" >>>
=== RESTART: C:\Users\User\AppData\
N=5
4
8
0
index= 3
>>>
  
```

Рис. 2. Результат программы

Использование инструкции break в цикле с параметром for

На рисунке 3 представлена программа использования инструкции **break** в цикле с параметром **for**. По программе видно, что использование инструкции **break** в цикле **for** мало чем отличается от использования в цикле **while**. При выполнении кода программы получаем результат, показанный на рисунке 2.



```

4.5.2.py - C:/Users/User/App...
File Edit Format Run Options Window Help
N=int(input ('N='))
p=0
for i in range(1,N+1):
    x=int(input())
    if x == 0:
        p=i;
        break
print('index=',p)
Ln:9 Col:2
  
```

Рис. 3. Код программы (for)

Инструкция **break** используется для выхода из цикла, когда получен ожидаемый результат выполнения программы. Рассмотрите, как будет выглядеть графическая блок-схема инструкций **break** и **continue** (схемы 3 и 4).



Схема 3. Блок-схема инструкции **break**

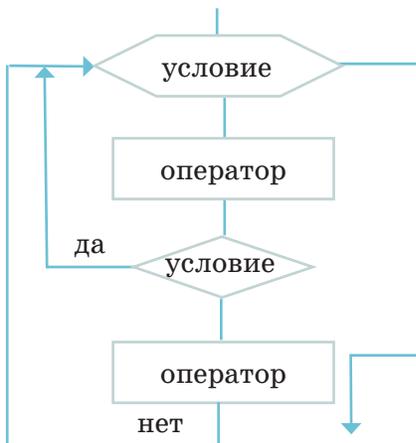


Схема 4. Блок-схема инструкции **continue**



Анализ



Проанализируйте графические блок-схемы **break** и **continue**, а также коды программ, в которых они были использованы. Сравните графические блок-схемы инструкций с программой. Как создать блок-схемы другим способом?



Синтез



Составьте и представьте полную блок-схему задания для приведенного выше примера.



Оценка



Оцените инструкции **break** и **continue**, сравнивая основные сходства и различия в выполняемой функции. Какие существуют преимущества использования этих инструкций для облегчения разработки программы? Всегда ли можно получить ожидаемый результат, используя только одну из этих двух инструкций? Объясните причины.



Вопросы для размышления

1. Как инструкция **break** влияет на работу цикла?
2. Сколько раз выполнится цикл **for i in range(1, 10):**
`if i ==4:`
`break?`
3. Какие результаты мы получим, если **break** в цикле использовать без конкретного условия?
4. Можно ли использовать несколько инструкций **break** в цикле? Обоснуйте причины.



Задание



Задание №1. Дано положительное целое число N . Если какая-либо цифра заданного числа равна нулю, то остановите цикл и в результате выведите «Error», в противном случае отобразите само число в обратном порядке.

Например	Результат
570256	Error
4589	9854



Задание №2. Дана строка S , состоящая из латинских букв и запятой «,». Создайте блок-схему алгоритма и программу, которая отображает на экране буквы до запятой из данной строки и их количество.

Например	Результат
Alma, almurt	Alma 4



Задание №3. Дано положительное целое число N . Создайте блок-схему, которая находит наименьшую из цифр этого числа. Напишите программный код задачи.

Например	Результат
78856954	4



Домашнее задание

Проанализируйте программные коды 1 и 2, предложенные ниже. Каким будет результат на экране, когда вы выполните программу?

```

1  s = 0
   for i in range(11, 21):
       s = s + i % 10
       if s > 15:
           break
   print (s)
    
```

```

2  x = 10; s = 0
   while x > 5:
       x = x - 2
       s = s + x
       if s > 20:
           break
   print (x)
    
```



Как используется инструкция по управлению циклом `else` в программировании на языке Python?



Подумай

- Почему важно пользоваться инструкцией по управлению циклом `break` и `continue` при написании программы?
- Какие функции выполняет команда `else` оператора проверки условия?



Новые знания

Вы уже познакомились с использованием инструкций `break` и `continue` в двух видах цикла. Эти инструкции использовались в теле цикла совместно с оператором проверки условий. Еще одной из таких инструкций является инструкция по управлению циклом `else`. В отличие от предыдущих двух инструкций `else` размещается снаружи циклов `for` и `while`. Инструкция `else` показывает, что цикл полностью выполнил все шаги и закончил свою работу в нормальных условиях или остановился сразу через `break`. Инструкция `else` выполняется только в том случае, если цикл полностью закончил свою работу, не используя `break`.

```
for i in range(k, n, x):
    <оператор 1 или выражение >
    <оператор 2 или выражение >
    ...
    <оператор n или выражение>
else:
    оператор или выражение>
```

Схема 1. Полная структура цикла `for`

```
while <условие>:
    <оператор 1 или выражение >
    <оператор 2 или выражение >
    ...
    <оператор n или выражение>
else:
    оператор или выражение>
```

Схема 2. Полная структура цикла `while`

На схеме 1 представлена полная структура цикла `for`. Если в теле цикла нет инструкции `break`, то цикл полностью выполняет все возможные шаги в своей работе. Только после этого выполнится выражение или оператор в блоке `else`. А теперь обратим внимание на полную структуру цикла `while` на схеме 2. Этот цикл начинает проверку условий цикла. Если условие истинно, то цикл продолжает работать.



иначе – айтпесе – `else`
тело цикла – цикл денесі – `loop body`

А если условие ложно, то выполняются команды блока **else**, а оператор цикла завершает работу. В схеме 3 представлена блок-схема алгоритма цикла **while**. Блок-схема алгоритма цикла **for** аналогична этой структуре. Рассмотрим практические задания для понимания работы циклов в случае их использования совместно с блоком **else**.

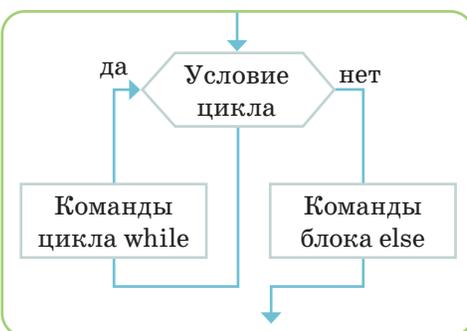


Схема 3. Блок-схема алгоритма цикла **while**



Практическая работа

Практическое задание №1. Дана последовательность целых чисел длиной $N(1 \leq N \leq 20)$. Напишите программу, отражающую следующее условие. Если последовательность состоит только из положительных чисел, то выведите на экран сумму всех чисел последовательности. В противном случае выдайте ответ «Обнаружено отрицательное число».

1) Например	Результат	2) Например	Результат
N=5	S= 23	N=5	Обнаружено отрицательное число
2		2	
3		3	
5		5	
6		4	
7		-3	

Пояснение. Для создания программы задания используем цикл **for** вместе с блоком **else**. Ввод чисел последовательности помещаем в тело цикла. Код программы задания представлен на *рисунке 1*. При выполнении кода программы получаем результат, показанный на *рисунке 2*.

```

4.6.2.py - C:/Users/User/AppData/Local/Pro...
File Edit Format Run Options Window Help
N=int(input ('N='))
s=0
for i in range(1,N+1):
    x=int(input())
    s+=x
    if x <0:
        print('В последовательности имеется отрицательное число')
        break
else:
    print('s=',s)
Ln:6 Col:10
    
```



Рис. 1. Код программы



```

Python 3.7.4 Shell
File Edit Format Run Options Wind
Python 3.7.4 (tags/v3.7.4:e093591
Type "help" "copyright", "credi
>>>
=== RESTART: C:\Users\User\AppData
N=5
2
3
5
6
7
S= 23
>>>
    
```

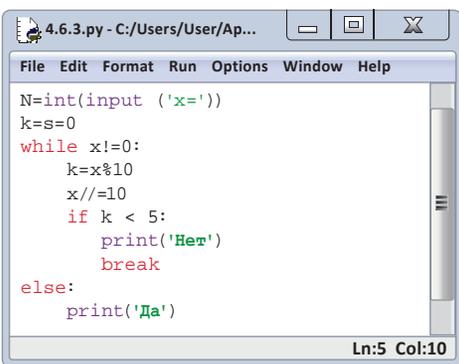
Рис. 2. Результат программы

Практическое задание №2. Дано натуральное число X . Если цифры, составляющие заданное число, не меньше 5, то такое число называют «сильным числом».

Если данное число – «сильное число», то на экране выводится «Да», а иначе – «Нет». Составьте программу.

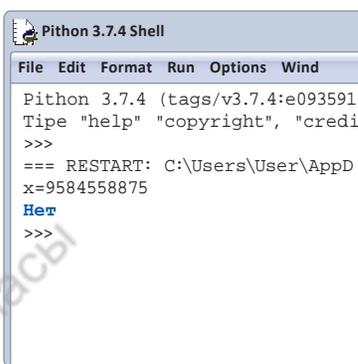
Например	Результат
59867	Да
977479	Нет

Пояснение. Для создания программы задания мы используем цикл **while** вместе с блоком **else**. Достаточно выделить цифры числа путем определения остатка и сравнить с 5. Код программы задания представлен на *рисунке 3*, результат показан на *рисунке 4*.



```

4.6.3.py - C:/Users/User/App...
File Edit Format Run Options Window Help
N=int(input ('x='))
k=s=0
while x!=0:
    k=x%10
    x//=10
    if k < 5:
        print('Нет')
        break
else:
    print('Да')
Ln:5 Col:10
  
```

```

Python 3.7.4 Shell
File Edit Format Run Options Wind
Python 3.7.4 (tags/v3.7.4:e093591
Type "help" "copyright", "credi
>>>
=== RESTART: C:\Users\User\AppData
x=9584558875
Нет
>>>
  
```



Рис. 3. Код программы

Рис. 4. Результат программы



Анализ



Выполните анализ кода программы практических заданий 1 и 2 (*рис. 1, 3*). Поясните назначение блока **else** в двух программах.



Синтез



Составьте и представьте блок-схему алгоритма цикла **for**.



Оценка



Сравните и найдите основные сходства и различия инструкций **break**, **continue**, **else** цикла. Какова значимость этих инструкций в разработке программы? Насколько облегчится создание программы?

**Задание**

*

Задание №1. Дано положительное целое число N . Создайте программу, в которой, если число «7» имеется среди цифр этого числа, то цикл остановится и выведется на экран «Error», в противном случае отобразится сумма цифр заданного числа.

Например	Результат
92576	Error
123456	21

**

Задание №2. Задана строка L , состоящая из латинских букв. Определите, встречается ли буква 'k' в этой строке. Разработайте программу, которая выведет на экран «yes», если в строке имеется данная буква, в противном случае ответом будет «no».

Например	Результат
uhuyhukgggff	yes

**

Задание №3. Дана последовательность длиной N ($1 \leq N \leq 20$), состоящая из целых чисел и одного дробного числа. В некоторых случаях последовательность может и не содержать дробное число. Создайте программу, которая отображает позицию дробного числа, если в заданной последовательности есть дробь, а в противном случае выводит ответ «no». Данные в программе должны вводиться вертикально.

Например	Результат
6 45 69 327 32.5 87 23	Index = 4

**Домашнее задание**

Выполните проект по предложенной теме.

Тема проекта	Направления и вопросы для исследования
Управление циклом в языке программирования Python	<ul style="list-style-type: none"> • Виды и назначение циклов. • Особенности работы с циклическими алгоритмами. • Способы использования инструкций управления continue, break, else



Как исправить ошибку, постепенно проверяя работу программы на языке **Python**?



Подумай

- Каково значение алгоритма для программирования?
- Как определить и исправить ошибки, допущенные в программе? Расскажите, опираясь на свой опыт.
- В чем необходимость постепенного выполнения строк программы?



Новые знания

Чем лучше программист знает операторы языка программирования и особенности их использования, тем меньше ошибок он допускает. Но, тем не менее, даже опытный программист может допустить ошибки.

На любом языке программирования можно своевременно редактировать и корректировать (**отладка**) программу, написанную на языке программирования. Отладка программы (**debugging**) – это обнаружение, локализация и устранение ошибок в компьютерной программе. Язык программирования имеет собственные средства исправления ошибок. На своем языке программисты называют ошибки, допущенные в программе, **багами** (от англ. *bug* – «технический дефект»), корректировку кода программы – **дебаггерами** (отладчики) (от англ. *debugger* – «корректор»). В языке **Python** существуют собственные средства редактирования, дающие возможность отлаживать программу, исправляя ошибки.

При работе с ошибками в программе возникают следующие вопросы.

- Как определить ошибки в программе?
- Как правильно исправить ошибки?



трассировка алгоритма – алгоритмнің трассировкасы – tracing algorithm
отладка – түзету – debugging

Ошибки, которые может допустить программист, условно делятся на три группы:

- 1) синтаксические;
- 2) времени выполнения;
- 3) алгоритмические.

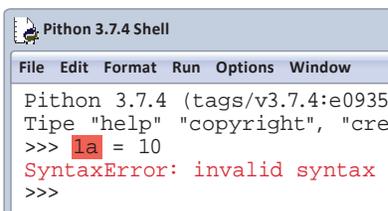
Во время работы программы ее команды выполняются последовательно друг за другом с помощью процессора компьютера. Программист не может точно определить, какая команда выполняется в конкретный момент времени выполнения программы. Чтобы узнать причину неправильной работы программы, необходимо знать четкий порядок выполнения команд. Его называют *трассированием*, или трассировкой программы. Алгоритм трассировки представляет собой порядок работы со встречающимися ошибками.

Трассировка – это процесс выполнения программы постепенно, шаг за шагом (step-by-step).

Во время трассировки программист может, задавая команды, выполнить программу последовательно строка за строкой.

Синтаксическая ошибка

В ошибке на *рисунке 1* имя переменной начинается с цифры. Программа сразу обнаруживает такие ошибки во время интерпретации. Если ошибки не исправлены, программа перестает работать. Если ошибки исправлены, то переходим к исправлению следующей ошибки. И так шаг за шагом исправляем выявленные в программе ошибки.



```
Python 3.7.4 Shell
File Edit Format Run Options Window
Python 3.7.4 (tags/v3.7.4:e0935
Type "help" "copyright", "cre
>>> 1a = 10
SyntaxError: invalid syntax
>>>
```

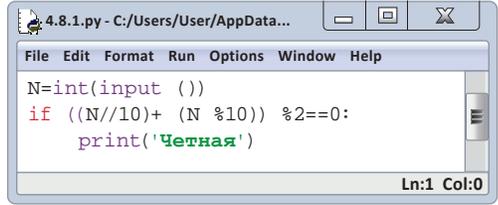
Рис. 1. Синтаксическая ошибка

Ошибки времени выполнения программы

Еще один вид ошибок – ошибка времени выполнения программы (runtime errors). Эта ошибка проявляется при исполнении программы во время запуска или во время тестирования. Например, причиной возникновения этой ошибки может служить случай, когда программа пытается прочитать текстовый файл input.txt, который не существует в компьютере. Эта ошибка указывается после того, как файл не был найден. Если в программе произошла ошибка, то она останавливается, выдав соответствующее сообщение.

Алгоритмическая ошибка

Тестирование – это ввод подготовленных данных в компьютер для проверки работоспособности программы. Если программа правильно выполнила одну из ситуаций, предусмотренных при тестировании, и показала ошибку в другой, то эта ошибка является ошибкой составления алгоритма задачи. Например, использовать знак «+» вместо «-» или знак «*». Этот тип ошибки не виден при интерпретации, и программа будет запущена. Но будет выдаваться ошибочный результат. Для этого необходимо пересмотреть и отредактировать код программы. Процесс исправления ошибок можно назвать **отладкой**. Рассмотрим пример.



```
4.8.1.py - C:/Users/User/AppData...
File Edit Format Run Options Window Help
N=int(input ())
if ((N//10)+ (N %10)) %2==0:
    print('Четная')
Ln:1 Col:0
```



Рис. 2.

Алгоритмическая ошибка

Дано двузначное число. Разработаем проект, определяющий, четной или нечетной является сумма цифр заданного числа.

Код программы прошел интерпретацию и работает. Но данный код может определить только четную цифру. В коде программы не рассмотрена ситуация, когда сумма нечетная. Вот это и есть алгоритмическая ошибка (рис. 2).

Чтобы изменить программу и записать строку «else: ('Нечетная')» в случае невыполнения условия, необходимо исправить код программы, чтобы получить правильный результат при обнаружении нечетной суммы. Кроме того, в процессе программирования существует несколько различных ошибок, встречающихся чаще всего. (табл. 1).

Таблица 1. Типы ошибок

№	Код программы	Тип ошибки	Сообщение на экране при выполнении программы
1	<code>s = input() if s % 2==0: print('Четный')</code>	Применение действия, не соответствующего типу величины	TypeError: not all arguments converted during string formatting
2	<code>k=int(input()) p=0;t=k/p print(t)</code>	Число нельзя делить на нуль. Деление переменной на нуль	ZeroDivisionError: division by zero

Продолжение таблицы

3	<code>a = 5 print (a + b)</code>	Неизвестная величина <i>b</i>	NameError: name 'b' is not defined
4	<code>int("Hi")</code>	Преобразование строки в число	ValueError: invalid literal for int() with base 10: 'Hi'
5	<code>k=int(input()) t=input() print(k+t)</code>	Сложение целого числа и строки. Например, 45+'85'	TypeError: unsupported operand type(s) for +: 'int' and 'str'

Использование команды `quit()` при трассировке для исправления ошибки в программе является наиболее эффективным. Контроль работы тела цикла дает хороший результат при использовании совместно с оператором условия.

Как выполняется трассировка программы? Рассмотрим следующий пример.

Создайте программу, которая из *n* заданных чисел находит количество четных чисел. Определите результат с помощью трассировки, отслеживая программу шаг за шагом. Программный код примера показан на *рисунке 3*, а трассировка программы дана в *таблице 2*.

```
n = int (input ())
s = 0
for i in range (0, n):
    t = int (input ())
    if t % 2 == 0:
        s + = 1
print (s)
```

Рис. 3. Программный код примера

Таблица 2. Трассировка программы

Переменные	Условие цикла	Переменная	Условие	Переменная
n	i	i < n	t % 2 == 0	s
3	0	0 < 3 (да)	5 % 2 == 0 (нет)	0
	1	1 < 3 (да)	16 % 2 == 0 (да)	1
	2	2 < 3 (да)	4 % 2 == 0 (да)	2
			Результат	2



Анализ

На основании опыта создания предыдущих программ сгруппируйте ошибки, допущенные вами при разработке и выполнении программы. Проведите их анализ в соответствии с ошибками в *таблице 1*. Приведите примеры.



**Вопросы для размышления**

1. Исходя из своего опыта, назовите ошибки, с которыми сталкивались в процессе программирования.
2. Что такое трассировка алгоритма?
3. Как осуществляется трассировка программы?
4. В чем значимость использования трассировки программы?

**Задание**

*

Задание №1. Напишите программу, которая находит сумму цифр чисел между k и p . Какую ошибку вы допустили при выполнении задания? Проверьте работу операторов, расположенных в теле цикла, путем трассировки.

Например	Результат
97 103	61

*

Задание №2. Определите ошибки в программном коде, предложенном ниже.

1

```
a = input('a=')
s = 0
for i in range(1, 10):
    s = s + i
    a = a * i
print(s/a)
```

2

```
p = int(input('p='))
s = 0; i = 0
while i < p:
    i = i + 1
    s = s + i
t = s * k
print(t)
```

**

Задание №3.

Разработайте программу, отображающую на экране все трехзначные числа до заданного числа N ($100 \leq N \leq 999$), состоящие из разных цифр в разрядах.

Например	Результат
107	102 103 104 105 106 107

Примечание. Трехзначные числа начинаются со 100.

**Домашнее задание**

Составьте задачу для решения одним из видов цикла с параметром **for** или цикла с условием **while**. Напишите программу для составленной задачи и наберите ее на компьютере. Выполните трассировку программы, которую вы написали.

Раздел IV

Программирование алгоритмов на языке программирования Python

4.8–4.9

Учимся программированию. Итоговые практические задания



Как при решении задачи эффективно использовать циклы при разработке ее программы в Python?



Подумай

- Какие имеются особенности в использовании оператора цикла **while**?
- Какие ошибки чаще всего допускают в использовании оператора цикла **for**?
- Насколько применение инструкций управления циклами **continue**, **break**, **else** повышает эффективность разработки программ?
- Какие существуют эффективные стороны постепенной проверки выполнения программы?



Практическая работа



Примерные задания для понимания путей программирования
Задание №1. Прямоугольная таблица. Разработайте программу, которая выводит на экран прямоугольную таблицу размером $N \times N$, где N ($1 < N < 20$). Образец выполнения задания приведен на *рисунке 1*. Проанализируйте реализацию вывода в программе и составьте блок-схему.

Например	Результат			
4	1	2	3	4
	5	6	7	8
	9	10	11	12
	13	14	15	16

```

4.8.1.py - C:/Users/User/App...
File Edit Format Run Options Window Help
N=int(input())
p=0
for i in range(1,N+1):
    for j in range(1,N+1):
        p+=1
        print(p,end="t")
    print()
Ln:6 Col:22
  
```

Рис. 1. Код программы



палиндром числа – палиндром сан – number palindrome
 разработка программы – программа әзірлеу – program development
 последовательность чисел – сандар тізбегі – sequence of numbers
 трехзначное число с разными цифрами – әр түрлі цифрлары бар үш таңбалы сан – three-digit number with different digits



Задание №2. Палиндром числа

Составьте программу, которая найдет и выведет на экран палиндромы среди чисел от А до В ($1 \leq A, B \leq 10^{18}$). Если в заданном промежутке нет палиндрома, то на экран выведите «-1». Палиндромом

№	Например	Результат
1	1800 2020	1881 1991 2002
2	1800 1880	-1

называется число, которое одинаково читается слева направо и справа налево. Пример выполнения задания приведен на *рисунке 2*. Выполните анализ строк вывода в программе.



Задание №3. Трехзначные числа с разными цифрами

Определите общее количество трехзначных чисел с разными цифрами (используя вложенные циклы). Пример выполнения задания представлен на *рисунке 3*. Проанализируйте результаты вывода программы.

```

File Edit Format Run Options Window Help
A, B=map(int,input().split())
t=0
for i in range(A,B+1):
    n=i; s=0
    while n > 0:
        k=n%10; n//=10
        s=s*10+k
    if s==i:
        t=t+1; print(s, end=' ')
else:
    if t==0:
        print(-1)
Ln:9 Col:8
    
```

Рис. 2. Код программы задания №2

```

File Edit Format Run Options Window Help
p=0
for i in range(1,10):
    for j in range(0,10):
        for k in range(0,10):
            if i!=j and j!=k and i!=k:
                p+=1
print(p)
Ln:9 Col:0
    
```

Рис. 3. Код программы задания №3

Опираясь на приведенные выше образцы программ, выполните задания.



Задание



Задание №1. Одинаковые цифры.

Даны двузначные числа в промежутке от К до N ($10 \leq K, N \leq 99$). Отобразите на экране нечетные двузначные числа, состоящие из двух нечетных цифр в заданном промежутке.

Например	Результат
15 32	15 17 19 31



Задание №2. Различные цифры. Даны четырехзначные числа в промежутке от M до N ($1000 \leq M$, $N \leq 9999$). Отобразите на экране количество чисел из заданного промежутка, состоящих из различных цифр. Если в этом интервале нет такого числа, выведите на экран «-1».

№	Например	Результат
1	1245 1256	9
2	1221 1229	-1



Задание №3. Наибольшая площадь. Стороны N ($1 \leq N \leq 20$) четырехугольников заданы длиной a ($1 \leq a \leq 20000$) и шириной b ($1 \leq b \leq 20000$). Выведите на экран площадь и порядковый номер четырехугольника с наибольшей площадью.

Например	Результат
3	30
4 3	2
6 5	
9 2	



Задание №4. Прямоугольная таблица. Выведите на экран прямоугольную таблицу размерностью $N \cdot N$, где N ($1 < N < 20$), по представленному образцу.

Например	Результат
4	0 1 1 1 1 0 1 1 1 1 0 1 1 1 1 0



Задание №5. Треугольная таблица. Выведите на экран треугольную таблицу размерностью $N \times N$, где N ($1 < N < 20$), по представленному образцу.

Например	Результат
5	2 3 4 5 3 4 5 4 5 5



Задание №6. Цифра 4. Определите, сколько раз цифра 4 встречается в числах промежутка от K до N . Если цифра 4 не встретилась ни разу, то выведите на экран «-1». Составьте блок-схему задачи и напишите программный код. Используйте вложенный цикл с инструкцией `else`.

№	Например	Результат
1	4444 4450	21
2	1055 1063	-1

* Задание №7. Цифры 2 и 5

Из заданного N -значного числа ($1 \leq N \leq 10^9$) составьте число, состоящее только из цифр 2 и 5, расположенных в произвольном порядке. Если эти цифры не имеются, то отобразите ответ «0». Составьте блок-схему задачи и напишите программный код. Используйте инструкцию `continue`.

№	Например	Результат
1	54522158	52255
2	14789631	0

** Задание №8. Общие цифры

Даны числа N ($1 \leq N \leq 10^9$) и M ($1 \leq M \leq 10^9$). Найдите цифры, которые встречаются в двух заданных числах. Если у них нет общей цифры, выведите «-1». Напишите программный код задачи. Используйте вложенный цикл.

Например	Результат
125360 267894	62

** Задание №9. Палиндром

Определите, является ли заданное число N ($1 \leq N \leq 10^{18}$) палиндромом. Разработайте программу. Если число является палиндромом, то выведите в ответе «Да», иначе – «Нет».

№	Например	Результат
1	2332	Да
2	12324	Нет

** Задание №10.

Последовательность чисел

Даны числа от K до N ($1 < K, N < 10^9$). Выведите на экран из заданного промежутка числа, сумма цифр которых является четной.

Например	Результат
30 40	31 33 35 37 39 40

Алгоритм выполнения задания:

- внимательно прочитать задание и понять условие;
- создать математическую модель задачи;
- разработать алгоритмическую блок-схему;
- разработать и осуществить трассировку программы.



Как эффективно использовать теоретические и практические знания по языку программирования **Python** при выполнении проектных работ?

**Подумай**

- Каковы правила выполнения проектных работ?
- Какие проекты для использования в повседневной жизни можно выполнить посредством программирования?

**Задание**

Выберите одну из предложенных тем проекта. Выполните индивидуально или в паре, представьте его. В ходе выполнения проекта, кроме изучения теоретического материала, важным является разработка программных продуктов.

**Тема проекта №1**

«Язык программирования Python и преимущества программирования в нем»

Краткое содержание. Описывая историю языка программирования Python, следует отметить, чем он отличается от других языков программирования. Необходимо отразить особенности языка программирования Python, важность его использования.

Тема проекта №2

«Типы циклов в языке программирования Python и их программирование»

Краткое содержание. Необходимо подробно рассмотреть циклические алгоритмы и остановиться на особенностях при их программировании. Как циклы используются в других языках программирования и чем они отличаются друг от друга?

Тема проекта №3**«Создание простой тестовой программы на языке программирования Python»**

Краткое содержание. Изучите способы создания тестов в Python. Объясните особенности разработанных тестовых образцов. Представьте разработанный программный продукт, сделайте акцент на его достоинствах и недостатках. Создайте презентацию.

Тема проекта №4**«Решение прикладных задач с помощью программирования»**

Краткое содержание. Прикладная математика, т.е. изучение способов решения задач с использованием метода программирования задач, связанных с применением математики в реальной жизни. Разработка небольших программных продуктов. Представление презентации.

Тема проекта №5**«Создание электронных учебников с использованием языков программирования»**

Краткое содержание. Изучение способов решения задач прикладной математики, т.е. математики, встречающейся в реальной жизни, и их реализация с помощью программирования. Разработка небольших программных продуктов. Представление презентации.

Тема проекта №6**«Программирование физических задач и лабораторных работ на Python»**

Краткое содержание. Разработка и презентация программы, которая автоматически рассчитывает заданные физические величины по формулам при решении задач по физике. Представление презентации.

Тема проекта №7**«Программирование геометрических задач»**

Краткое содержание. Изучение вычисления различных параметров геометрических фигур (площади, периметра, объема) путем программирования. Разработка небольших программных продуктов. Представление презентации.

Рекомендации по эффективной защите проекта

- Представьте выбранную тему.
- Объясните выбор темы.
- Сформулируйте цели и задачи.
- Опишите, какие использовали средства.
- Сделайте краткий обзор темы.
- Поделитесь своими мыслями о пользе реализации проекта.
- Расскажите о новых знаниях и навыках, полученных при работе над проектом.
- Представьте результаты проекта.
- Выслушайте мнения и оценки аудитории.

Критерии успешного проекта

- Самостоятельная реализация проекта.
- Актуальность темы.
- Раскрытие темы.
- Свой способ решения проблемы.
- Ораторское мастерство при защите проекта и умение держать себя на публике.
- Умение использовать технические средства и визуальные средства.
- Умение правильно и точно отвечать на поставленные вопросы.

Каким должен быть отчет о выполненном проекте

- Почему я выбрал эту тему?
- Что узнал нового, чему научился?
- Как осуществлял поиск необходимой информации?
- Как редактировал информацию?
- Как совмещал теорию и практику?
- Какие части проекта получились наиболее эффективными?
- Что не смог выполнить?
- Что было самым легким в проекте?
- Какие трудности возникали во время работы над проектом?
- Если бы я начал проект заново, что бы изменил в нем?
- Мое впечатление от работы над проектом.

Раздел V Практическое программирование

5.1 Постановка проблемы



Как работать с постановкой проблемы в создании моделей задач на языке программирования Python?



Подумай

- Что вы знаете о циклах? Какое отношение это понятие имеет к нашей жизни?
- Какова роль циклов в природе и жизни человека? Приведите примеры.
- Почему важно правильно уметь определить, сформулировать постановку проблемы при программировании?



Новые знания

В 7 классе вы уже познакомились с этапами решения задач. Первым и самым важным из этих этапов, то есть фундаментом для решения задачи, является этап постановки проблемы. Ведь если неправильно понять поставленную задачу, то решение будет ошибочным независимо от дальнейших действий.

С этапами выполнения поставленной задачи на языке программирования Python вы познакомились в младших классах. Как теперь реализовать их на практике с помощью программы? Рассмотрим проект задачи «Деление бревна на равные части» в Python. Причина, по которой мы считаем его проектом, заключается в том, что, даже если программный код этой задачи создается по-другому, он может дать правильный результат. Для этого рассмотрим алгоритм полного решения этой задачи как авторский проект. Полное выполнение проекта, этапы реализации программирования представлены в качестве практической работы. Нужно тщательно проанализировать этап выполнения данного проекта и понять, что создание программы тесно связано с математикой.



математическая модель проекта – жобаның математикалық моделі –
mathematical model of the project
тестирование данных – деректерді тестілеу – data testing



Практическая работа

Проект «Деление бревна на равные части»

Определите, можно ли бревно длиной N ($2 \leq N \leq 100$) распилить на M ($2 \leq M < 20$) равные части. Если невозможно, то разделите на части, длина которых отличается только на 1.



№	Например	Результат
1	24 6	4 4 4 4 4 4
2	13 4	3 3 3 4
3	37 8	4 4 4 5 5 5 5 5

1. Постановка проблемы. Правильное понимание условий задачи. В задаче определяются два условия.

Условие 1: разделение бревна длиной N метров на M равных частей.

Условие 2: в случае невозможности распилить на одинаковые части, необходимо разделить на целые части с разницей в длине на 1. Других вариантов в решении задачи нет, так при разделении бревна на части обязательно выполняется одно из этих двух условий.

2. Разработка математической модели проекта. В тесте 1 рассматриваемой задачи можно распилить бревно на равные части, если число N на M делится без остатка. То есть для выполнения условия 1 должно быть выполнено условие $N \% M = 0$. Если это условие не выполнено, то результат задачи будет выдан в соответствии с условием 2. Если посмотреть на представленные тестовые данные, то можно увидеть, что в решении задачи имеется своя закономерность. Например, рассмотрим тестовые данные №2. Деление 13 на 4 нацело в результате дало число 3 и остаток 1. В ответе следует вывести K ($K=3$) $M-P$ раз, т.е. 3 раза и $(K+1)$ в ответе P раз (рис. 1). Эту закономерность можно проверить и с другими тестовыми данными.

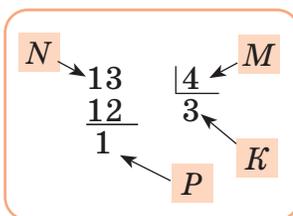


Рис. 1. Разработка математической модели проекта

При невозможности разделения бревна на равные части по первому условию можно сразу распилить на целые части по второму условию с разницей не более чем на единицу.

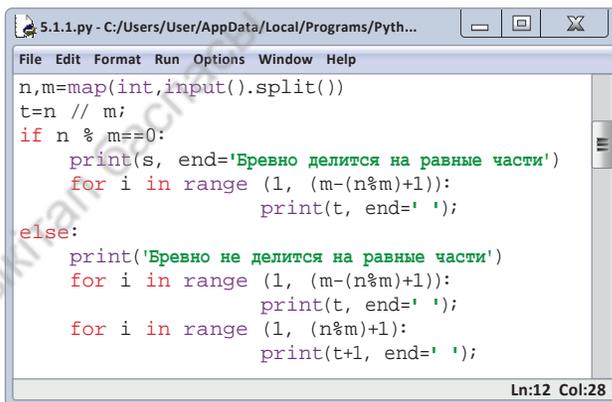
3. Разработка алгоритма. Вы нашли основную закономерность задания и создали математическую модель. Для того чтобы разработать программу для этой задачи, необходимо систематизировать и разделить на шаги. Вы составляете последовательность команд, которые будут выполняться на каждом шаге.

Алгоритм решения задачи.

- Выбор операторов ввода и вывода результата. В нашем случае используем операторы **input** и **print**.
- Чтобы проверить основное условие задачи $N \% M = 0$, необходимо воспользоваться оператором проверки условий.
- Чтобы вывести последовательность чисел, используем оператор цикла с параметром.

4. Преобразование алгоритма на язык программирования. Напишем программу проекта по составленному алгоритму (рис. 2). При написании программы нужно постоянно обращать внимание на то, что каждый оператор, используемый в программе проекта, должен поддерживать совместимую работу друг с другом.

5. Ввод и тестирование данных. После разработки программы проекта запускаем его в действие. При запуске проекта появится форма проекта, представленная на рисунке 2. При



```

5.1.1.py - C:/Users/User/AppData/Local/Programs/Pyth...
File Edit Format Run Options Window Help
n,m=map(int,input().split())
t=n // m
if n % m==0:
    print(s, end='Бревно делится на равные части')
    for i in range (1, (m-(n%m)+1)):
        print(t, end=' ');
else:
    print('Бревно не делится на равные части')
    for i in range (1, (m-(n%m)+1)):
        print(t, end=' ');
    for i in range (1, (n%m)+1):
        print(t+1, end=' ');
Ln:12 Col:28
  
```

Рис. 2. Код программы

вводе данных следует обратить внимание на то, что не должны вводиться данные, за исключением ограничений, указанных в условии задачи. Если вводить в программу в качестве вводных данных 0 и 2 или 10 и 16, программа покажет ошибку. В первом случае длина бревна не может быть равна нулю. Такое бревно невозможно разделить на части. Во втором случае не выполняется условие деления бревна на целые части по условию задачи.

6. Получение результатов задачи и его анализ. После запуска программы по результатам тестирования при получении правильного результата на все заранее подготовленные тестовые данные считается, что задача решена на 100 процентов. Если результаты, полученные в результате тестирования, не соответствуют условиям задачи, то по-

иск логических, алгоритмических ошибок, имеющих в структуре программы, следует проверить методом трассировки. На *рисунке 2* видно, что при вводе данных, приведенных в задаче 3, в программу выдается верный ответ. Можно проверить работу программы не только с предложенными данными, но и создать другие данные тестирования задачи.



Анализ



Сравните этапы решения задач в математике, физике и информатике. Определите сходство и различие.



Синтез



Выберите одну из известных вам задач по математике. Составьте и представьте проект программы с использованием этапов решения выбранной задачи. Дайте название проекту.



Оценка



Исследуйте, есть ли другие виды решения задач на компьютере. Какие имеются различия между этапами решения задач на компьютере, которые вы изучили на уроках, и другими видами, с которыми вы ознакомились самостоятельно?



Вопросы для размышления

1. Из каких этапов состоит решение задачи в среде программирования?
2. На какие проблемы следует обратить внимание при выборе компонента проекта?
3. Какие условия следует учитывать при составлении тестовых данных для задачи и тестирования?
4. Какие специфические особенности решения задач на компьютере вы можете назвать? Приведите примеры.



Задание

В среде программирования создайте программы для следующих заданий, опираясь на этапы решения задач.

*

Простая задача

Даны числа A , B и знаки арифметических действий, которые должны быть выполнены с ними. Например, $+$, $-$, $/$, $*$. Создайте программу, которая выполняет одно из этих действий с числами A , B в зависимости от знака операции.

Например	Результат
$7 + 9$	16

**

Среднее арифметическое положительных чисел

Дано N ($0 < N \leq 100$) вещественных чисел. В следующей строке заданы N вещественных чисел, значения которых не превосходят по модулю 100. Найдите среднее арифметическое положительных чисел. Выведите среднее арифметическое положительных чисел с двумя десятичными знаками. Если отсутствуют положительные числа, выведите сообщение «Нет» (без кавычек).

Например	Результат
4	2.00
3 -9 1 2.0	
5	Нет
-1 -7 -19 -20 -40.0	

**

Посадка деревьев

Одним из важных направлений развития Республики Казахстан является охрана природы. С целью улучшения экологической ситуации в стране, формирования у молодежи патриотизма и бережного отношения к богатствам родного края в республике действует программа «Жасыл Ел». В ходе ежегодных республиканских акций «Подари родному городу дерево» и др. школь-



Например	Результат
$N = 5$	4
200	
157	
362	
965	
145	

ники г. Нур-Султана совместно с родителями и общественными деятелями высаживают деревья. В этом году ученики 8-го класса разделились на N бригад и начали активно сажать деревья ($1 \leq N \leq 20$). Если известно количество деревьев, посаженных каждой бригадой, напишите программу, которая определит порядковый номер победившей бригады при посадке деревьев.

**

Баскетбольный матч

В рамках реализации Государственной программы по развитию физической культуры и спорта в Республике Казахстан, а также с целью популяризации баскетбола в разных городах страны регулярно проходят турниры среди школьников. Баскетбольный матч состоит из 4-х этапов. Входными данными являются целые числа a и b ($0 \leq a, b \leq 100$), которые указывают количество очков, заработанных командами на каждом этапе. Создайте программу, которая определяет победителя на основе результатов игры между двумя командами. В результате игры должен быть получен ответ «команда №1», «команда №1» или «равная игра».



Например	Результат
26 17	1-я команда
13 15	
19 11	
14 16	

Количество заданных цифр в числе

В первой строке записано целое число N ($N < 10^{18}$). Число N может быть только положительным. Во второй строке дана цифра K . Подсчитайте количество цифр K в числе N .

Например	Результат
15265425522 5	4

**

Результат программы

Определите результат программы после выполнения следующего фрагмента программы:

①

```
p=1; i=3
while i <= 9:
    p+=(i / 2)
    i = i + 1
print (p)
```

②

```
s=0; i=7
while i > 1:
    s+=i % 2
    i-= 1
print (s)
```

③

```
S=100; K=0
for i in range(1, 6) :
    K+=i*i
    S-=K
print (S)
```

Раздел V Практическое программирование

5.2 Разработка алгоритма



Как составить простые алгоритмы задач с использованием циклических алгоритмов в Python?



Подумай

- Какие сложности возникают при создании алгоритмов для задач по теме циклы?
- Какие распространенные алгоритмы, созданные с использованием циклических алгоритмов, вам известны?



Новые знания

В 7 классе вы познакомились с методами разработки алгоритмов ветвления. Теперь рассмотрим методы разработки циклических алгоритмов и решения задач с их использованием при программировании. По сравнению с линейными и разветвленными алгоритмами циклические более сложные. Перед разработкой программы необходимо уточнить постановку проблемы в соответствии с условием задачи и пройденной темой. На следующем этапе необходимо уточнить все условия, которые будут рассмотрены для решения задачи. В целом разработка алгоритма задачи – это не только создание его графической блок-схемы. Важно то, что мы отражаем внутри этой блок-схемы: сколько и какие шаги (алгоритмы) должна включать программа по условиям задачи. На этом этапе мы должны четко понимать, с помощью какого вида алгоритма мы достигнем цели. При этом важную роль играет математическая модель задачи. Поэтому необходимо использовать знания, полученные на уроках математики. Если вы любую задачу можете перевести на язык математики и создать ее математическую модель, то написанная вами программа будет правильной. Поэтому в процессе программирования помогает знание наиболее часто используемых алгоритмов.



создание алгоритма – алгоритмді құру – algorithm creation
эффективный алгоритм – тиімді алгоритм – efficient algorithm

В связи с чем возникла необходимость знания наиболее часто используемых готовых алгоритмов?

Вы уже знаете, что в процессе написания программы поставленную задачу можно разделить на несколько подзадач и рассмотреть их отдельно. Основная часть задач, рассматриваемых в процессе программирования, выражается в виде задач для определения эффективности (оптимизации). В таких задачах обязательно следует определять наибольшее, наименьшее, самые тяжелые, самые легкие и т.д. Для этого лучше использовать заранее проверенные, эффективные алгоритмы. Они позволяют, во-первых, экономить время, во-вторых, дают возможность решить главную задачу с учетом всех рассматриваемых условий. Остановимся на нескольких стандартных алгоритмах, знакомых многим программистам. Некоторые из них известны и вам. В чем необходимость их использования?

Одним из важных аспектов алгоритма является его скорость. В процессе программирования перед разработчиком постоянно стоит проблема экономии рабочего времени программы. Например, на тему «вложенные циклы» выполнили задачу по выводу на экран простых чисел из заданной последовательности чисел. Для определения простых чисел был использован цикл `for` с параметром до квадратного корня проверяемого числа. Почему? Остановимся на причинах. Возьмем, например, 50 как тестируемое число. Если необходимо определить, является это число простым или составным, то нужно разделить это число на все числа между 1 и 50. Чтобы проверить число 50, нужно выполнить цикл 50 раз. Чтобы проверить число большее, чем 1 000 000, необходимо выполнить тело цикла миллион раз. Это очень много, так как для выполнения цикла 1 раз требуется 1 секунда. Соответственно, миллион раз – миллион секунд.

```
for j in range(1, n):
```

Следовательно, разработанный алгоритм неэффективен. Должен быть другой эффективный алгоритм. Если учесть, что в этом примере обязательно деление числа 50 на числа от 1 до 50 и что делители любого числа могут быть равны половине того же числа, за исключением самого числа (например, делители 50 могут быть числа до 25, не считая 50), то в нем не остается необходимости разделить число до 50. В этом случае работа цикла сокращается в 2 раза и работает 25 раз, чтобы проверить, является ли число 50 простым.

```
for j in range(1, n //2+1):
```

Если использовать теорию чисел, то работа цикла может быть еще больше сокращена.

Теорема. Если простое число p является наименьшим делителем числа a , то выполняется условие $p \leq \sqrt{a}$.

Например, число 59 – это простое число. Его квадратный корень $\approx 7,6$ не делится ни на одно из простых чисел 2, 3, 5, 7. Таким образом, для проверки числа 59 также 2 ...7 достаточно 6-кратной работы цикла ($\sqrt{59} \approx 7$). Это в 8 раз меньше исходного решения. А если проверочное число 1 000 000, то цикл работает 1000 раз.

`for j in range (2,round(math.sqrt(x))):`

Проведя анализ одного примера, вы увидели, насколько важно знать эффективные алгоритмы при написании программы. Для изучения практического применения алгоритмов выполните задачу «Покраска забора» на компьютере.



Практическая работа

Покраска забора. Для покраски забора вокруг своего дома у Болат имеются краски пяти разных цветов. Чтобы определить, каким цветом ему красить, Болат нанес краски разных цветов на доски забора (рис. 1). В результате забор стал разноцветным. Болат хочет покрасить забор в один цвет, затрачивая при этом как можно меньше усилий. Помогите Болату решить его задачу. Известны количество досок в заборе Болат N ($N < 100$) и их цвет, один из пяти в интервале между 1 и 5. В качестве ответа на задачу должно вывестись наименьшее количество досок, подлежащих повторной покраске.

№	Например	Результат
1	$N=7$ 3 1 5 1 1 1 5	3

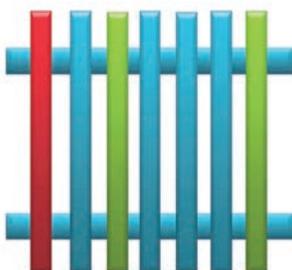


Рис. 1

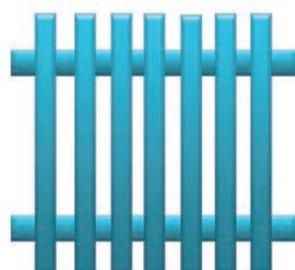


Рис. 2

Еще один вопрос, на который стоит обратить внимание, это условие задачи. В большинстве случаев задачи на программирование составляют на основе событий, встречающихся в жизни. Наиболее сложным является понимание условия таких задач и проблема их программирования. Например, приведенная выше задача, выраженная на математическом языке, содержала бы всего одну текстовую строку, несложную для понимания. «Найдите максимально повторяющееся число в последовательности и количество его повторений».

Остановимся на строке вывода задачи (рис. 3). Достаточно найти наибольшее количество повторений и исключить их из общего числа введенных.



Анализ



Проанализируйте программу задачи «Покраска забора» (рис. 3). При анализе подготовьте тестовые данные, содержащие различные ситуации. Тщательно проверяйте работу программы.



Синтез



Измените условие задачи так, чтобы она была связана с повседневной жизнью (в качестве образца возьмите задачу из практической работы).

Задача. Дана последовательность, состоящая из N чисел. Для этой последовательности составьте программу, которая находит сумму членов последовательности до P -го места ($1 \leq P, N < 100$).



Домашнее задание

Дана последовательность, состоящая из N чисел. Найдите сумму членов этой последовательности, которые делятся на K без остатка ($1 \leq K, N < 100$).

```

5.2.1.py - C:/Users/User/AppData/Local/Programs/P...
File Edit Format Run Options Window Help
z=0; z1=0; z2=0; z3=0; z4=0
n=int(input())
for i in range(1, n+1):
    x=int(input())
    if x==1:
        z+=1
    elif x==2:
        z1+=1
    elif x==3:
        z2+=1
    elif x==4:
        z3+=1
    elif x==5:
        z4+=1
max=z
if z1>max:
    max=z1
if z2>max:
    max=z2
if z3>max:
    max=z3
if z4>max:
    max=z4
print(n-max)
Ln:25 Col: 0
  
```



Рис. 3. Код программы

5.3– 5.4 Программирование алгоритма. Практическая работа



Как составить программу решения задачи с использованием циклических алгоритмов в Python?



Подумай

- Какие особенности циклического алгоритма вы можете назвать?
- Какие трудности могут возникнуть в программировании циклических алгоритмов?



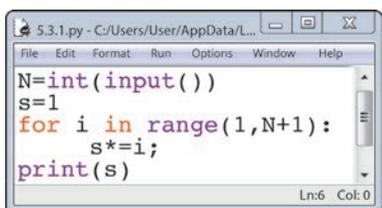
Новые знания

В младших классах в процессе программирования задач вы уже познакомились с путями создания алгоритмических блок-схем. Теперь, составив блок-схему задач на использование цикла, изучим практику преобразования ее на язык программы. Для этого рассмотрим ряд задач с созданием блок-схемы и познакомимся с путями их решения.

Задача 1. Вычисление N ($1 \leq N \leq 20$) факториала. Создайте блок-схему и преобразуйте ее в программу.

Математическая модель задачи. Факториалом числа N называется произведение всех чисел между 1 и N . Как знак факториала, используется символ «!». Например, $5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120$. Другими словами, нужно найти произведение чисел от 1 до N .

Блок-схема задачи представлена на схеме 1. Переводим этот алгоритм на язык программирования (рис. 1).



```

5.3.1.py - C:/Users/User/AppData/L...
File Edit Format Run Options Window Help
N=int(input())
s=1
for i in range(1,N+1):
    s*=i;
print(s)
Ln:6 Col:0
  
```



Рис. 1. Код программы

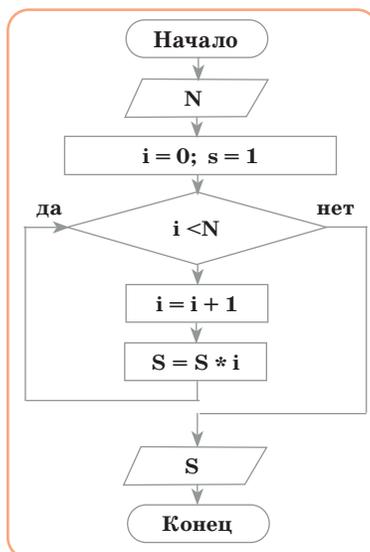


Схема 1. Блок-схема

Задача 2. Дано число n ($1 \leq n \leq 10^{12}$). Выведите на экран наибольшую и наименьшую цифры, составляющие это число. Составьте блок-схему задачи и преобразуйте ее в код программы.

Например	Результат
85486569	max=9 min=4

```

5.3.2.py - C:/Users/...
File Edit Format Run Options
Window Help
n=int(input('n='))
s=1; max=0; min=10;
while n>0:
    k=n%10
    n=n//10
    if k>max:
        max=k
    if k<min:
        min=k
print('max=',max)
print('min=',min)
Ln:11 Col:17
    
```

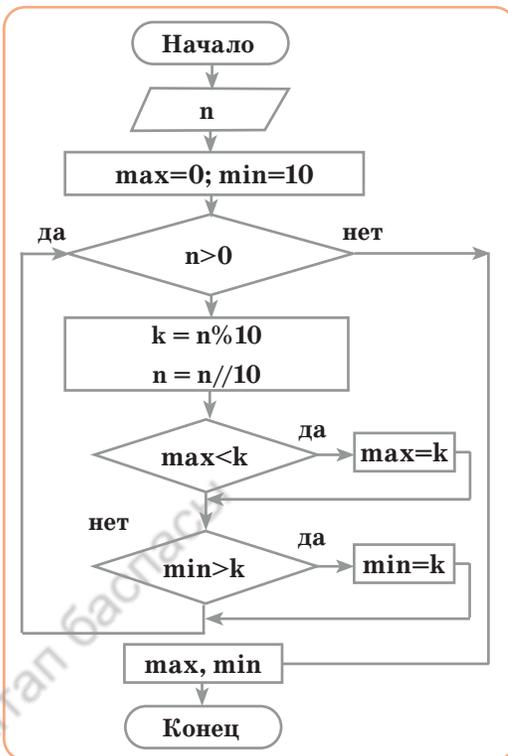


Рис. 2. Программа задачи 2

Схема 2. Блок-схема задачи 2

Математическая модель задачи. Математическая модель данной задачи проста. Необходимо только выделить цифры заданного числа n и проверить их по алгоритму определения наибольшего и наименьшего значения. Для решения задачи используем цикл проверки условий.



Анализ



Выполните анализ программного кода задачи 2. Какие результаты будут получены при вводе значений 1) $n=495582$ и 2) $n=385845$? Проанализируйте полученные результаты и дайте пояснение.



- условие задачи – есептің шарты – conditions of the problem
- проверка кода программы – программа кодын тексеру – checking program code
- вычисление – есептеу – calculating

**Задание**

*

Задание №1. Составьте блок-схемы следующих программ.

```

1 x = int(input() )
  a,b = 0, 0
  while x>0:
    a = a + 1
    b = b + x%10
    x = x//100
  print(a)
  print(b)

```

```

2 s = 0
  for k in range(1, 51):
    if k %5=0:
      s = 2 * k + s
  print(a)

```

*

Задание №2. Что мы получим на экране, когда будут запущены следующие программные коды? Найдите ответ без проверки кода программы на компьютере.

```

1 s = 0
  n = 0
  while s < 51:
    s = s + 11
    n = n * 2
  print(n)

```

```

2 s = 175
  n = 0
  while s + n < 300:
    s = s - 5
    n = n + 25
  print(n)

```

*

Задание №3. Допущены ошибки в написании следующих программных кодов. Определите эти ошибки.

```

1 s = input()
  k = 7
  while s < 30
    s = s + 2 * k
    k: =k + 2
  print(k)

```

```

2 s = 0
  for k in range(1, 50):
    if k %7 = 0:
      s = s + k
  print('2 * s')

```

**

Задание №4. Сумма четных чисел. Дана последовательность чисел длиной N ($1 \leq N \leq 1000$). Найдите сумму последних цифр четных чисел в последовательности. Последовательность должна состоять из натуральных чисел, не превышающих 10^9 . Ввод данных осуществляется вертикально.

Например	Результат	Пояснение
5 425 4 658 12 35	14	$4 + 8 + 2 = 14$

**

Задание №5. Выделение дробной части числа

Дана последовательность действительных чисел длины N ($1 \leq N \leq 1000$). Выведите на экран дробную часть всех чисел последовательности с точностью 0,001. Последовательность должна состоять из действительных чисел, не превышающих 10^5 . Создайте блок-схему программы. Ввод данных осуществляется вертикально.

Например	Результат
5 4.12312 3 45.01 2.1 7.00005	0.123 0.000 0.010 0.100 0.000

**

Задание №6. Поиск. Дано число X ($1 \leq X \leq 10000$). Найдите ближайшее к нему простое число. Если заданное число является простым, то в ответе выдайте его. Если есть два таких числа, то выведите наименьшее.

№	Например	Результат	Пояснение
1	31	31	Тест 2: простые числа, близкие к 15, равны 13 и 17. Ответ по условию задачи – 13.
2	15	13	

**

Задание №7. Найти число последовательности

Последовательность Фибоначчи – это последовательность чисел 1, 1, 2, 3, 5, 8, 13, каждое из которых равно сумме двух предыдущих слагаемых, начиная с третьего.

Формула последовательности Фибоначчи: $F_0 = 0, F_1 = 1, F_n = F_{n-1} + F_{n-2}, n \geq 0$. Найдите число в X -й позиции этой последовательности. ($1 \leq X \leq 50$). Последовательность: 0, 1, 1, 2, 3, 5, 8, 13.

№	Например	Результат
1	4	3
2	11	89



Домашнее задание

Дано число N ($1 \leq N \leq 10^{12}$). Напишите программу, которая определяет, сколько раз цифра 7 встречается в этом числе.

№	Например	Результат
1	14785747	3
2	786	1

5.5– 5.6 Тестирование программы



Как проверить эффективность работы программы тестированием в Python?



Подумай

- Что такое тестирование программы?
- Какие типы тестов для проверки программы вам известны?



Новые знания



Тестирование – это испытание, проверка правильной работы программы или ее отдельных частей. В процессе тестирования проверяется работоспособность программы, не содержащей явных ошибок. Оно не зависит от того, какой вид алгоритма (линейный, разветвленный или циклический) используется в программе. Тестовые данные должны обеспечить проверку всех возможных условий возникновения ошибок.

Процесс тестирования состоит из одинаковых этапов во всех алгоритмах. В нем можно выделить три этапа:

1. Проверка при обычных условиях.
2. Проверка в экстремальных (граничных) условиях.
3. Проверка в исключительных условиях.

После создания программы для заданной задачи переходим к ее выполнению. В 7 классе вы изучили способы проверки работоспособности программы с помощью образцов тестовых данных, подготовленных заранее. Причина, по которой программа может не пройти отборочные тесты, связана с ошибками в разработке программы.

Ошибки, встречающиеся в программировании

Следует помнить, что абсолютно на всех этапах разработки программы могут быть допущены ошибки, среди которых: неверная по-



синтаксические ошибки – синтаксистік қателер – syntax errors

логические ошибки – логикалық қателер – logical errors

проверка граничных условий – шектік (шегаралық) шарттарды тексеру – check boundary conditions

проверка в исключительных условиях – ерекше жағдайлар үшін тексеру – check in exceptional conditions

становка задачи; неправильно разработанный алгоритм решения; неучтенные обстоятельства, при которых могут быть допущены логические, семантические и синтаксические ошибки; ошибки, допускаемые при выполнении операции и отправке данных; ошибки ввода-вывода и др.

Обычно синтаксические ошибки выявляются при трансляции. Транслятор не может обнаружить множество других ошибок, так как не знает конечную задачу программиста. Следовательно, отсутствие синтаксических ошибок не является достаточным условием корректности программы.

Трансляция программы – преобразование программы из одного языка программирования в другой. Транслятор – программа или техническое средство, выполняющее трансляцию программы.

Примеры синтаксических ошибок:

- пропущенные знаки препинания;
- неправильно поставленные скобки;
- некорректная запись оператора;
- неправильная запись служебного слова;
- некорректная запись имени переменной;
- отсутствие окончания условия цикла;
- отсутствие описания массива и т.д.

Ошибки, которые не может обнаружить транслятор

Существует множество ошибок, которые не могут быть обнаружены транслятором, хотя те операторы, которые используются в программе, записаны правильно. Приведем примеры таких ошибок.

Логические ошибки:

- неправильное указание пути перехода к выполнению ветви алгоритма несмотря на то, что правильно составлено условие проверки;
- неполный учет возможных условий;
- пропущены один или несколько значений, необходимых для выполнения блоков алгоритма.

Ошибки, встречающиеся в циклах:

- не указано начало цикла;
- некорректно составлено условие окончания цикла;
- неправильно определено число итераций;
- бесконечный цикл.

Ошибки ввода-вывода и ошибки при работе с данными:

- неправильно описаны типы данных;
- некорректная организация ввода необходимого количества данных;
- неправильная корректировка данных.

Ошибки, допускаемые при использовании переменных:

- использование переменных без указания их начальных значений;
- некорректное использование одной переменной вместо другой.

Ошибки, допускаемые при работе с массивами:

- массивы не приведены заранее к нулю;
- неправильное описание массивов;
- массивы с некорректным указанием индексов.

Ошибки, допускаемые при выполнении арифметических операций:

- неправильное использование типа переменной (например, вместо целой переменной используется вещественная переменная);
- неточно указан порядок алгебраических действий;
- деление на нуль;
- извлечение квадратного корня отрицательного числа;
- потеря разряда числа.

Все эти ошибки можно выявить на этапе тестирования программы.

**Практическая работа**

Рассмотрим задачу «Делители» (задача из районной олимпиады по информатике) для ознакомления с тестированием программы.

Делители

Дано натуральное число b . Требуется определить его делители. Если данное натуральное число b делится без остатка на число a , то число a является делителем числа b . Например, у числа 15 есть 4 делителя: 1, 3, 5, 15. Для каждого данного числа b определите, четным или нечетным будет количество его делителей. Число b дано в количестве N ($1 < N \leq 105$). Для каждого из чисел b ($1 \leq b \leq 10^{18}$) выведите значение 0 – если будет четное количество делителей, и 1 – если количество делителей будет нечетным. Время ответа программы на каждый тест не превышает двух секунд.

Например	Результат	Комментарий
5 5 8 16 11 9	0 0 1 0 1	У числа 8 всего 4 делителя: 1, 2, 4, 8. Количество делителей – четное число. У числа 16 всего 5 делителей: 1, 2, 4, 8, 16. Количество делителей – нечетное число.

Перейдем к образцам тестовых данных (табл. 1), составленных к этой задаче. Такие тесты разрабатываются и представляются до решения задачи. Всего дано 10 тестов, но мы остановимся на 5-ти основных. Потому что тесты одинакового типа повторяются по 2 раза.

Таблица 1. Образцы тестовых данных для задачи

№	Входные данные	Выводимый результат
1	10 1 2 49 77 49 55 80 94 49 77	1 0 1 0 1 0 0 0 1 0
2	10 830 585 868 529 683 740 967 557 828 1147	0 0 0 1 0 0 0 0 0 0
3	10 501139 786769 976039 635209 902903 977269 557993 635209 654481 876831	0 1 0 1 0 0 0 1 1 0
4	10 5000207 1524071 50000214792889 50000348753931 50001431026237 50001060703878 50001009723014 50000483494043 50000950188029 50000119016331 50000702267865	0 1 0 0 0 0 0 0 0 0
5	10 500000001283061852 500000001394837861 499999937510787169 500000000548349143 500000000709382267 499999937510787169 500000002022608063 500000000107748979 500000000076465733 500000000770135069	0 0 1 0 0 1 0 0 0 0

Рассмотрим тестовые данные, приведенные в таблице 1.

Тест №1: для обычных условий.

Тест №2: среди чисел встречается одно число с нечетным количеством делителей.

Тест №3: для случаев, когда граница данных 10^6 .

Тесты №№4, 5: проверочные тесты значения b на соответствие числам в степени 10^{18} . Как видно, даны тестовые данные на обычные, граничные и экстремальные условия задачи.

Почему такие тестовые данные составлены для проверки решения задачи? Какие трудности возникают при решении задачи?

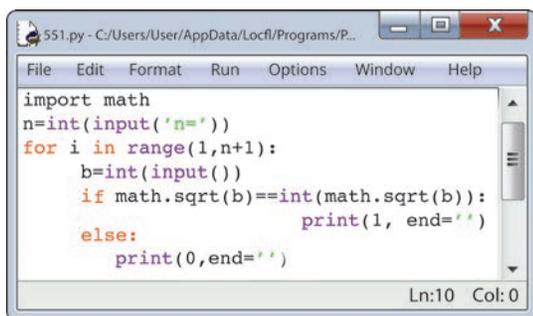
Чтобы найти делители одного числа в решении задачи, необходимо использовать один цикл с параметром, а для того, чтобы найти делители последовательности чисел, – еще один цикл. Тогда для решения этой задачи необходимо будет использовать 2 вложенных цикла.

А если учесть, что длина последовательности передаваемых чисел составляет 100 000, то программа не может ответить на каждый тест в течение 2 секунд. Здесь возникает вторая проблема: ограничение числа $b - 10^{18}$. При определении делителя числа вы будете делить его на числа до его половины. Так что вы не сможете организовать цикл до половины этого числа. По алгоритму, который вы знаете, можно определить, что задача позволит пройти только 20 процентов тестовых заданий. Следовательно, необходимо рассмотреть другой эффективный алгоритм решения этой задачи. В качестве эффективного алгоритма, который может быть использован для этого расчета, необходимо вычислить квадратный корень из заданного числа. Если число извлекается из корня целым, то количество делителей этого числа является

нечетным числом, в противном случае – четным числом.

Например, квадратный корень из 36 равен 6. Количество его делителей 9: 1, 2, 3, 4, 6, 9, 12, 18, 36. Корень квадратный из 40 равен 6,324... Количество его делителей 8: 1, 2, 4, 5, 8, 10, 20, 40.

Тогда можно представить программу задачи в простой форме (рис. 1).



```

551.py - C:/Users/User/AppData/Local/Programs/P...
File Edit Format Run Options Window Help
import math
n=int(input('n='))
for i in range(1,n+1):
    b=int(input())
    if math.sqrt(b)==int(math.sqrt(b)):
        print(1, end='')
    else:
        print(0, end='')
Ln:10 Col:0
  
```



Рис. 1. Код программы задачи «Делители»



Анализ



Проанализируйте 3 примерных теста, разработанных для кода программы задачи «Делители». Введите следующие тестовые данные, запустив программу. Определите, какой результат будет выведен на экран.

① 5

16 81 256 100 318

② 5

11 8 26 101 910

③ 5

64 3 17 10 121



Вопросы для размышления

1. Какие этапы включает тестирование программы?
2. Какие виды ошибок можно определить в ходе тестирования?
3. Какой набор тестов выдается на тестирование программы?
4. Какое влияние оказывает тестирование на работу программы?
5. Почему важно тестировать программу?



Задание



Задание №1. Определите, какие результаты тестирования будут получены на основе образцов тестов без ввода программного кода (рис. 2) в компьютер.

```
s = int(input() )
n = 0
while 2 * s < 150:
    s = s + 5
    n = n + 2
print(n)
```

№	Образцы тестов	Результат
1	0	
2	5	
3	15	
4	7	
5	100	

Рис. 2



Задание №2. Дана последовательность, состоящая из 0 и 1. Найдите самую длинную последовательность единиц. Используя этапы тестирования, подготовьте тест, который включает 5 различных ситуаций, чтобы определить правильность решения задачи.

Например	Результат
01001111011	4



Задание №3. Подготовка деталей. Рабочие должны подготовить N деталей за 5 часов. Они сделали X деталей в первый час работы и тратят

№	Например	Результат
1	7 12 45	12
2	3 10 60	7

T минут на подготовку каждой из оставшихся деталей. X , N и T ($X \leq N \leq 15$, $T \leq 240$). Определите, сколько деталей рабочие подготовят за 5 часов. Подготовьте 5 тестов для проверки правильности решения задачи, охватывающих различные ситуации. Напишите программу.



Домашнее задание



Стоимость книги

Дано количество книг – N ($N \leq 20$) и цена одной книги – K . Выведите на экран вычисленную стоимость книг.

Цена одной книги составляет K тенге, цена 2 книг – $2 \cdot K$ тенге ... и т.д.

Например	Результат
3 500	Книга 1 – 500 тенге Книга 2 – 1000 тенге Книга 3 – 1500 тенге

Раздел V Практическое программирование

5.7 Практические задания. Подготовка к олимпиаде



Как в Python составить проект, создать модель задачи и записать код программы?



Подумай

- Какие процессы, происходящие в повседневной жизни, можно показать через программирование?
- Какие ошибки часто допускают пользователи при написании программы?
- Какие знания необходимо сочетать в поиске алгоритма решения задачи?



Практическая работа

*

Задание №1. Сравнение

Даны два числа А и В. Произведение цифр какого числа будет наименьшим?

Например	Результат
121 43	121

*

Задание №2. Равенство нулю

Определите количество чисел между К и Р, у которых одна из цифр равна нулю ($1 < K < P < 1000$).

Например	Результат
8 22	2

*

Задание №3. Воздушные шары

Воздушные шары расположены в виде треугольника, в первом ряду – 1, во втором ряду – 2, в третьем ряду – 3 и т.д. Если задано N воздушных шаров, сколько рядов можно создать с их помощью. Во входном файле первая строка содержит только одно положительное целое число N ($1 < N < 999$). Выходной файл содержит количество строк, которые могут быть заполнены шарами.

Input.txt	Output.txt
15	5



равно нулю – нөлге тең – is equal to zero

Объяснение решения задачи. Если количества заданных шаров недостаточно для формирования полного ряда, то мы считаем ряд неполным. Например: если дано 18 шаров, то ответ на задачу должен быть 5, потому что трех шаров не хватает, чтобы полностью заполнить следующий ряд.

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 ...

*

Задание №4. Спортсмен

Спортсмен начал готовиться к соревнованию по бегу через B дней. В 1-й день он пробежал дистанцию K метров. Если он увеличивает дистанцию бега на M процентов каждый день, какую дистанцию он пробежит в последний день тренировки? Ответ округлите до целого значения.

Например	Результат
11 1000 10	2593

**

Задание №5. Бой часов

Старинные часы отбивают время каждые полчаса. Причем в начале каждого часа количество ударов совпадает со временем на часах (в 3 часа – 3 удара, в 12 часов дня и в 12 часов ночи – 12 ударов и т.д.) И еще 1 раз они бьют в середине каждого часа. Предложите программу, которая просматривает заданный интервал времени в течение дня и определяет, сколько раз часы бьют в течение этого времени. Для ввода даны числа a , b и c , d .

A – начальный час, b – начальная минута, c – начальный последний час, d – последняя минута ($0 \leq a, c \leq 23, 0 \leq b, d \leq 59, b \neq 30, d \neq 30$). Пример проекта показан на рис. 1.



Рис. 1. Иллюстрация к заданию №5

Например	Результат
5 20 10 25	45
10 25 5 20	135

**

Задание №6. Игра «Бараны и козлы»

Марат и Еркин любят играть в разные логические игры. На этот раз они выбрали игру «Быки и коровы», но переименовали персонажей игры в честь героев своего любимого мультфильма «Қошқар мен теке» (рис. 2). Правила игры «Бараны и козлы» представлены ниже. Марат загадывает 4-значное число, состоя-

щее из разных цифр. Еркин должен найти это число шаг за шагом под руководством Марата. Для числа, загаданного Маратом, Еркин отгадывает, сколько баранов и сколько козлов. Среди скрытых цифр, загаданных Маратом, бараны – это те цифры, которые совпали и находятся на своем месте. Если отгаданная цифра верна, но не находится в нужной позиции, то эта цифра – козлы.

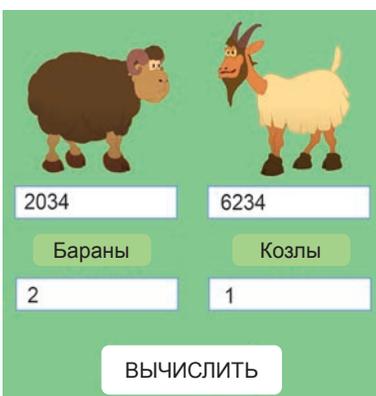


Рис. 2. Иллюстрация к заданию №6

Например	Результат
5671 7251	1 2
1425 1425	4 0



Задание №7. Определение тока и сопротивления в проводнике
 Большое значение имеет применение закона Ома в разделе физики «Постоянный ток». На *рисунке 3* показана схема последовательного и параллельного соединения сопротивления проводника. В соответствии с формулой закона Ома для части цепи разработайте программу, которая вычисляет ток, напряжение, сопротивление, находит общее сопротивление, полученное путем подключения резисторов последовательно и параллельно (*рис. 4*). Для выполнения задачи мы используем формулу закона Ома для части цепи.

Формула закона Ома: $I = \frac{U}{R}$

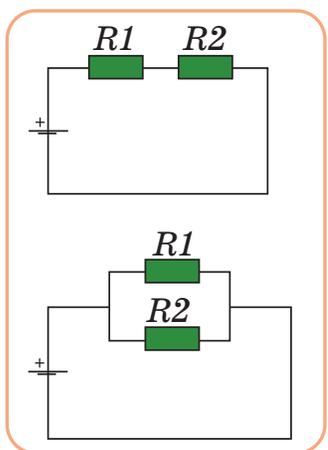


Рис. 3. Последовательное и параллельное соединения



Рис. 4. Иллюстрация к заданию №7

СЛОВАРЬ – СӨЗДІК – DICTIONARY

Русский	Казахский	Английский
А		
абсолютная ссылка	абсолютті сілтеме	absolute reference
алфавитный метод	алфавиттік әдіс	alphabetic method
анализ данных	деректерді талдау	data analysis
аналитический	аналитикалық	analytical
арифметико-логическое устройство	арифметикалық-логикалық құрылғы	arithmetic and logic unit
асимметричная цифровая абонентская линия	асимметриялық цифрлық абоненттік желі	asymmetric digital subscriber line
аспекты использования компьютера	компьютерді пайдаланудың аспектілері	aspects of using computer use
Б		
байт	байт	byte
бит	бит	bit
блок питания	қоректендіру блогы	power supply unit
блок-схема	блок-схема	flowchart
В		
вероятностный метод	ықтималдық әдісі	probabilistic method
вес символьной информации	символдың ақпараттық салмағы	weight of symbol information
видеокарта	бейнекарта	video adapte, video card
визуальный	көрнекі	visual
вложенный цикл	кірістірілген цикл	embedded loop
внутреннее устройство компьютера	компьютердің ішкі құрылымы	internal computer structure
время передачи	тасымалдау уақыты	delivery time
вставить	кірістіру	insert

встроенные функции	кіріктірілген функциялар	built-in functions
выполнение цикла	циклды орындау	loop execution
вычисление	есептеу	calculating
Г		
график линейной функции	сызықтық функция графигі	graph linear functions
Д		
дата и время	мерзім мен уақыт	date and time
двоичный алфавит	екілік алфавит	binary alphabet
двоичный язык кодирования	екілік кодтау тілі	binary coding language
диаграмма	диаграмма	diagram
динамический ряд	динамикалық қатар	dynamic series
доступность	қолжетімділік (оңтайлылық)	availability
Е		
если	егер	if
Ж		
жесткий диск	қатқыл диск	HDD (hard disk drive)
З		
звуковая карта	дыбыстық тақша	sound card
И		
И	және	and
измерение информации	ақпаратты өлшеу	measurement of information
или	немесе	or
иначе	әйтпесе	else
инженерный	инженерлік	engineering
инструкция цикла	цикл нұсқаулығы	loop instruction

интернет	интернет	internet
интерфейс	интерфейс	interface
информационная безопасность	ақпараттық қауіпсіздік	information security
информационный вес	ақпараттық салмақ	information weight
истина	ақиқат	true
К		
квадратное значение	квадрат мәні	square value
кибербуллинг	кибербуллинг	cyberbullying
количество информации	ақпарат саны	amount of information
коммутационный шнур	коммутатор бауы	switch cord
компьютерная сеть	компьютерлік желі	computer network
конфиденциальность	құпиялық	confidentiality
корреляция	корреляция	correlation
кэш-память	кэш-жад	cache memory
Л		
линейный	сызықтық	linear
логарифмический	логарифмдік	logarithmic
логические	логикалық	logic
логические ошибки	логикалық қателер	logical errors
ложь	жалған	false
М		
математическая модель проекта	жобаның математикалық моделі	mathematical model of the project
математическое выражение	математикалық өрнек	mathematically expression
материнская плата	аналық тақша	motherboard
многочлен	көпмүше	polynomial
мощность алфавита	алфавиттің қуаттылығы	power of the alphabet

Н		
надежность	сенімділік	reliability
негативный аспект	жағымсыз аспекті	negative aspect
неопределенность знаний	білімнің белгісіздігі	uncertainty of education knowledge
О		
облачная технология	бұлттық технология	cloud technology
облачные вычисления	бұлттық есептеулер	cloud computing
объем информации в тексте	мәтіннің ақпараттық көлемі	the amount of information in the text
оперативная память	жедел жад	emergency memory
отладка	түзету	debugging
П		
палиндром числа	палиндром сан	number palindrome
переход к началу цикла	циклдің басына бару	go to the beginning of the cycle
периферийные устройства	қосымша құрылғылар	peripheras
персональный компьютер	дербес компьютер	personal computer
порты ввода-вывода	енгізу-шығару порттары	input-output ports
последовательность чисел	сандар тізбегі	sequence of numbers
предложения работы или заработка	жұмыс немесе ақша табуды ұсыну	work or earnings offer
прерывание цикла	циклдың үзілуі	conditional break
привод оптических дисков	оптикалық дискжетектер	odd optical disk drive
провайдер	провайдер	provider
проверка в исключительных условиях	ерекше жағдайлар үшін тексеру	check in exceptional conditions

проверка граничных условий	шектік (шегаралық) шарттарды тексеру	check boundary conditions
проверка кода программы	программа кодын тексеру	checking program code
прогноз, прогнозирование	болжам, болжамдау	forecast, prediction
пропускная способность передачи канала	арнаның өткізу жылдамдығының қабілеті	capacity band width of channe`s transmission
пропускная способность сети	желінің өткізу қабілеті	network`s capcity bandwidth
процессор, микропроцессор	процессор, микропроцессор	processor, microprocessor
Р		
разработка программы	программа әзірлеу	program development
разрядность процессора	процессордың разрядтылығы	processor`s structure
регистр	регистр	register
сетевая безопасность	желідегі қауіпсіздік	network security
сетевая плата	желілік тақша	network card
С		
синтаксические ошибки	синтаксистік қателер	syntax errors
системная шина	жүйелік шина	system bus
слот расширения	слот кеңейтілімі	expansion slot
создание алгоритма	алгоритмді құру	algorithm creation
сообщение	хабар	message
сопроцессор	сопроцессор	coprocessor
относительная ссылка	салыстырмалы сілтеме	relative reference

ссылка	сілтеме	reference
ссылки и массивы	сілтемелер мен жиымдар	references and arrays
статика	статика	statics
статистический	статистикалық	statistical
степень	дәреже	degree
Т		
текстовые	мәтіндік	text
телo цикла	цикл денесі	loop body
тестирование данных	еректерді тестілеу	data testing
трассировка алгоритма	алгоритмнің трассировкасы	tracing algorithm
трехзначное число с разными цифрами	әр түрлі цифрлары бар үш таңбалы сан	three-digit number with different digits
У		
управление циклом	циклды басқару	cycle control
условие цикла	цикл шарты	loop condition
условие задачи	есептің шарты	conditions of the problem
устройство процессора	процессордың құрылысы	processor`s device
устройство управления	басқару құрылғысы	control device
Ф		
финансовые	қаржылық	financial
функции	функциялар	functions

Ц

целостность	тұтастық	integrity
цикл	цикл	cycle
цикл while («пока»)	while («өзірге») циклы	while loop

Ч

частота процессора	процессордың жиілігі	processor frequency
чип	чип	chip
число ядер	ядролардың саны	number of cores

Ш

шина	шина	bus
------	------	-----

Э

электромагнитное излучение	электрмагниттік сәуле	electromagnetic radiation
электронная таблица	электрондық кесте	spreadsheet
эффективный алгоритм	тиімді алгоритм	efficient algorithm
IP-адрес	IP-адрес	internet protocol address

ГЛОССАРИЙ

Абсолютные ссылки используются в формулах для указания фиксированного адреса ячейки. При перемещении или копировании формулы абсолютные ссылки не изменяются. Символ \$ используется, чтобы указать абсолютный адрес. Существуют два типа абсолютных ссылок: полная и частичная.

Алгоритм – это сформулированное правило, указывающее на действия, последовательное выполнение которых приводит от исходных данных к искомому результату.

Алфавит (гр. Ἀλφάβητος – письменная версия) – набор символов (букв, цифр и т.д.), из которых может состоять любое сообщение на данном языке.

Аспект (от лат. aspectus – «взгляд») – точка зрения, с которой рассматривается какое-либо явление, понятие, перспектива.

Булева функция – функция алгебры логики, аргументы которой принимают значения из двухэлементного множества (обычно $\{0,1\}$).

Вложенные циклы – это несколько простых циклов, которые используются один внутри другого.

Динамический ряд – совокупность однородных статистических величин, показывающих изменение какого-либо явления на протяжении определенного промежутка времени.

Защита информации – это комплекс мер, направленных на обеспечение информационной безопасности. На практике информационная безопасность означает поддержание целостности, доступности и, при необходимости, конфиденциальности информации и ресурсов, используемых для ввода, хранения, обработки и передачи данных. Таким образом, защита информации – это комплекс мер, предпринимаемых для предотвращения утечки, кражи, потери, несанкционированного удаления, изменения, подделки, несанкционированного копирования, блокировки. Он состоит из организационных, программных и технических методов и инструментов, направленных на удовлетворение ограничений безопасности.

Интернет-мошенничество – вид мошенничества с использованием интернета. Оно может включать сокрытие информации или представление неверной информации с целью вымогательства денег или имущества.

Интерфейс прикладной программы – 1) Набор задач (интерфейс), используемых программистами для написания прикладных программ в конкретной компьютерной среде. 2) Набор процедур, вызываемых прикладной программой для выполнения операций низкого уровня, выполняемых операционной системой.

Информационная безопасность – предотвращение несанкционированного доступа, использования, раскрытия, искажения, изменения, записи, уничтожения информации.

Кибербуллинг – это травля, оскорбления или угрозы, передаваемые жертве с помощью средств электронной коммуникации (сообщений в социальных сетях, мгновенных сообщений, электронных писем и sms).

Коммутатор (от англ. switch – переключатель) – устройство, предназначенное для соединения нескольких узлов компьютерной сети в пределах одного или нескольких ее сегментов.

Материнская плата (motherboard) является основной платой, на которой располагаются: центральный микропроцессор, сопроцессор, оперативная память, кэш-память, базовая система ввода-вывода (BIOS) и связанные с нею унифицированные разъемы (слоты, порты) для подсоединения других функциональных модулей и устройств. Главный функциональный компонент компьютера.

Микропроцессор является наиболее важным элементом на системной плате, который обрабатывает данные напрямую, в частности, выполняет арифметические и логические операции с выбранными данными. Микропроцессор – процессор, выполненный в виде одной либо нескольких взаимосвязанных интегральных схем.

Мошенничество – это приобретение чужого имущества путем кражи или злоупотребления доверием.

Мощность (размер) алфавита – это полное количество символов в алфавите. Мощность алфавита обозначается буквой N.

Операционная система (operating system) – это комплекс взаимосвязанных системных программ, назначение которого – организация взаимодействия пользователя с компьютером и выполнение всех программ. В функции операционной системы входят:

- осуществление диалога с пользователем;
- ввод-вывод и управление данными;
- планирование и организация процесса обработки программ;
- распределение ресурсов (оперативной памяти и кэша, процессора, внешних устройств);
- запуск программ на выполнение;
- всевозможные вспомогательные операции обслуживания;
- передача информации между различными внутренними устройствами;
- программная поддержка работы периферийных устройств (монитора, клавиатуры, дисковых накопителей, принтера и др.).

Отладка программы (debugging) – обнаружение, локализация и устранение ошибок в компьютерной программе.

Отладчик (debugger) – программа, предназначенная для анализа поведения другой программы, обеспечивающая ее трассировку (отслеживание и распечатку выполняемых программой команд, изменений переменных или данных о других событиях, связанных с выполнением программы), остановку в указанных точках или при выполнении указанных условий, просмотр и изменение ячеек памяти, регистров процессора и команд программы.

Параметр (от греч. *parametron* – «отмеривающий») в математике – величина, числовые значения которой позволяют выделить определенный элемент из множества элементов.

Прикладная программа (или приложение) – программа, предназначенная для выполнения определенных задач и рассчитанная на непосредственное взаимодействие с пользователем.

Прикладное (специальное) программное обеспечение – часть программного обеспечения, состоящего из отдельных прикладных программ и пакетов приложений, которые создают решения для прикладных проблем.

Прикладной программный интерфейс (application programming interface, API) – это описание способа, который позволяет какому-либо фрагменту программного обеспечения (ПО) обращаться к другой программе за получением сервиса. Этим сервисом может быть предоставление доступа к данным или выполнение конкретной функции.

Прогнозирование – специальное научное исследование конкретных перспектив дальнейшего развития какого-либо процесса. К основным методам прогнозирования относят:

- статистические методы;
- экспертные оценки (например, метод Дельфи);
- методы моделирования;
- интуитивные (то есть выполненные без применения технических средств, экспромтом, «в уме» специалистом, имеющим опыт применения научных методов в данном типе прогнозов).

Программная система – это такая система, в которую входит программное обеспечение. В общем случае программная система, помимо собственно программ, содержит еще и аппаратное обеспечение, а также обычно рассматривается в окружении других программно-аппаратных систем.

Системная программа – это программа, которая расширяет возможности операционной системы и управляет ресурсами компьютера: центральным процессором, памятью, вводом-выводом.

Системная шина – шина, предназначенная для передачи данных между периферийными устройствами и центральным процессором, или между периферийными устройствами и оперативной памятью.

Слот расширения – разъем на материнской плате, соединенной с системной шиной, который позволяет подключать к нему дополнительные платы, расширяющие конфигурацию устройства. Через слоты обычно подключаются видеокарты, звуковые карты, сетевые карты.

Сопроцессор – специализированный процессор, расширяющий возможности центрального процессора компьютерной системы, но оформленный как отдельный функциональный модуль. Физически сопроцессор может быть отдельной микросхемой или может быть встроен в центральный процессор. Различают следующие виды сопроцессоров:

- математические сопроцессоры общего назначения, обычно ускоряющие вычисления с плавающей запятой;

- сопроцессоры ввода-вывода (например, Intel 8089), разгружающие центральный процессор от контроля за операциями ввода-вывода или расширяющие стандартное адресное пространство процессора;
- сопроцессоры для выполнения каких-либо узкоспециализированных вычислений.

Статистика (от лат. *status* – «состояние дел») – отрасль знаний, наука, в которой излагаются общие вопросы сбора, измерения и анализа массовых статистических (количественных или качественных) данных; изучение количественной стороны массовых общественных явлений.

Теория вероятностей – наука, изучающая общие закономерности случайных явлений массового характера. Случайные явления характеризуются неопределенностью, сложностью и причинностью.

Тестирование программы (program testing) – проверка программы в рабочих условиях с некоторыми специально созданными тестовыми данными с целью определения ее работоспособности в соответствии с заданными критериями оценки.

Трассировка (trace) – метод проверки правильности функционирования программ при их выполнении путем отображения изменений всех значений переменных. Этим достигается сокращение поиска ошибочных команд, из-за которых переменные принимают неверные значения. Трассировка выполняется при помощи ввода в проверяемую программу специальных команд или с использованием специальных сервисных программ.

Функция (от лат. *functio* – «исполнение», «осуществление») в программировании – фрагмент программного кода (подпрограмма), к которому можно обратиться из другого места программы. Функция может принимать параметры и должна возвращать некоторое значение, возможно пустое. Функции, которые возвращают пустое значение, часто называют процедурами. Функция должна быть соответствующим образом объявлена и определена.

Цикл (cycle, loop) – разновидность управляющей конструкции в высокоуровневых языках программирования, предназначенная для организации многократного исполнения набора инструкций.

Цикл с условием while выполняется, пока истинно задающее его условие. Поэтому этот цикл также иногда называют циклом «пока». Часто цикл while используется, когда невозможно заранее предсказать, сколько раз необходимо выполнить тело цикла.

Цикл for в языке программирования Python предназначен для перебора элементов структур данных и некоторых других объектов.

Циклический алгоритм – это алгоритмическая структура, представляющая собой последовательность команд, которые повторяются до тех пор, пока не будет выполнено начальное условие. Написание программы с циклами упрощает процесс программирования.

Чипсет (англ. chipset) – набор микросхем, спроектированных для совместной работы с целью выполнения набора заданных функций. Так, в компьютерах чипсет, размещаемый на материнской плате, выполняет роль связующего компонента (моста), обеспечивающего взаимодействие

центрального процессора (ЦП) с различными типами памяти, устройства ввода-вывода, контроллерами и адаптерами ПУ, как непосредственно через себя (имея некоторые из них в своем составе), так и через другие контроллеры и адаптеры, с помощью многоуровневой системы шин.

BIOS (Basic Input/Output System – базовая система ввода/вывода) – это программа, которая выполняет первоначальный запуск персонального компьютера, настройку оборудования и поддержку функций ввода/вывода.

Работа BIOS заключается в обеспечении начальной загрузки компьютера с последующим запуском операционной системы, а также в хранении аппаратной конфигурации компьютера.

Изменяя настройки BIOS, можно управлять работой компонентов компьютера.

Функции программы:

- проверка наличия и работоспособности найденного оборудования;
- поиск, загрузка и запуск операционной системы;
- установка начальных параметров оборудования.

В ситуациях, когда ошибочно были внесены неверные параметры или произошел сбой настроек, BIOS можно вернуть к исходным параметрам (обнулить).

Основные виды программы: Award BIOS (от компании Phoenix Technologies) и Ami BIOS (от American Megatrends). Они отличаются по своей структуре.

Через меню доступна настройка следующих функций BIOS:

- системные дата и время;
- выбор устройства загрузки операционной системы (актуально при установке самой ОС);
- включение или отключение поддержки USB-устройств;
- просмотр значений или изменение напряжения, подающегося на процессор и модули RAM-памяти (для увеличения их производительности);
- установка пароля на просмотр настроек BIOS.

Break – инструкция, которая прерывает выполнение цикла. Управление передается на следующую инструкцию после тела цикла.

Continue – это инструкция, которая прерывает текущую итерацию цикла. Все инструкции после инструкции continue до конца блока цикла пропускаются.

IP-адрес (от англ. Internet Protocol Address) – уникальный идентификатор (адрес) устройства (обычно компьютера), подключенного к локальной сети или интернету.

Электромагнитные волны – процесс распространения электромагнитного поля в пространстве. Электромагнитная волна представляет собой процесс последовательного, взаимосвязанного изменения векторов напряженности электрического и магнитного полей, направленных перпендикулярно лучу распространения волны, при котором изменение электрического поля вызывает изменения магнитного поля, которые, в свою очередь, вызывают изменения электрического поля.

ИСПОЛЬЗОВАННАЯ ЛИТЕРАТУРА

1. *Корнеев В.Н., Куприянова А.В.* Microsoft Excel 2007. Просто о сложном. – М., 2008.
2. *Пташинский В.* Excel 2010 для начинающих. – М., 2013.
3. *Семенов В.П., Финкова М.А.* Excel 2013 на примерах. – СПб., 2016.
4. *Серогодский В.В., Финкова М.А., Прокди Р.Г.* Excel 2013. Пошаговый самоучитель + справочник пользователя. – СПб.: Наука и техника, 2016.
5. *Грицианова Т.Ю.* Информатика. Программирование в примерах и задачах. – М., 2016.
6. *Владимирова Н.А.* Увлекательная информатика. 5–11 классы. Логические задачи, кроссворды, ребусы, игры. – Волгоград, 2015.
7. *Паронджанов В.Д.* Учись писать, читать и понимать алгоритмы. Алгоритмы для правильного мышления. Основы алгоритмизации. – ДМК Пресс, 2016.
8. *Кормен Т.Х.* Алгоритмы. Вводный курс. – Вильямс, 2016.
9. *Андреева Е.В.* Программирование – это так просто. Программирование – это так сложно. Современный учебник программирования. – МЦНМО, 2018.
10. *Каракулев Ю.А., Иванов А.Н.* Руководство к решению задач с применением электронных таблиц EXCEL: Учебное пособие. – СПб., 2010.
11. *Карчевский Е.М., Филиппов И.Е., Филиппова И.А.* Excel 2010 в примерах. Учебное пособие. – Казань, 2012.
12. *Гульев И.А.* Компьютерные вирусы, взгляд изнутри. – М., 1998.
13. *Холмогоров В.* Компьютерная сеть своими руками. Самоучитель. – СПб., 2003.
14. *Шевелев Г.Е.* Информатика: Лабораторный практикум. – Томск, 2004.
15. *Киселев В.Г.* Практическая работа в Excel 2007. – Нижний Новгород, 2009.
16. *Горбачев А., Котлеев Д.* Microsoft Excel. Работайте с электронными таблицами в 10 раз быстрее. – ДМК Пресс, 2009.
17. *Сэнд У., Сэнд К.* Hello World! Занимательное программирование. – СПб., 2016.
18. *Мэттиз Э.* Изучаем Python. Программирование игр, визуализация данных, веб-приложения. – СПб., 2017.
19. *Хахаев И.А.* Практикум по алгоритмизации и программированию на Python. – ALT Linux, 2010.
20. *Седжвик Р., Уэйн К., Дондеро Р.* Программирование на языке Python. – М., 2017.
21. *Абрамян М. Э.* 1000 задач по программированию. Часть 1. – Ростов-на-Дону, 2004.

ССЫЛКИ

<https://pythonworld.ru/osnovy/cikly-for-i-while-operatory-break-i-continue-volshebnoe-slovo-else.html>

<http://programm.ws/page.php?id=686><http://programm.ws/page.php?id=686>ачать

<http://www.tvd-home.ru/prog/c6>

https://mediana.kz/kz/blog_companii/press_center/?title=cloud_

СОДЕРЖАНИЕ

Дорогие ученики!	3
РАЗДЕЛ I. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ КОМПЬЮТЕРА И СЕТЕЙ	
1.1. Измерение информации. Алфавитный подход	4
1.2. Измерение информации. Вероятностный подход.....	8
1.3. Внутреннее устройство компьютера и периферийные устройства	12
1.4. Процессор и его характеристики	19
1.5. Компьютерные сети	25
РАЗДЕЛ II. ЗДОРОВЬЕ И БЕЗОПАСНОСТЬ	
2.1. Негативные аспекты использования компьютера.....	30
2.2. Безопасность в сети.....	35
РАЗДЕЛ III. ОБРАБОТКА ИНФОРМАЦИИ В ЭЛЕКТРОННЫХ ТАБЛИЦАХ	
3.1. Статистические данные. Использование абсолютных и относительных ссылок.....	40
3.2. Использование абсолютных и относительных ссылок в решении практических задач	46
3.3. Встроенные функции	51
3.4. Использование встроенных функций при решении практических задач	56
3.5. Анализ данных на основе имеющейся информации	60
3.6–3.7. Решение прикладных задач.....	66
3.8. Проектные работы в MS Excel	74

РАЗДЕЛ IV. ПРОГРАММИРОВАНИЕ АЛГОРИТМОВ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON

4.1. Цикл for	77
4.2. Организация вложенных циклов.....	82
4.3. Цикл while	86
4.4. Управление циклом continue	90
4.5. Управление циклом break	94
4.6. Управление циклом else	98
4.7. Трассировка алгоритма	102
4.8–4.9. Учимся программированию.	
Итоговые практические задания	107
4.10. Проектные работы по программированию	111

РАЗДЕЛ V. ПРАКТИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

5.1. Постановка проблемы	114
5.2. Разработка алгоритма	120
5.3–5.4. Программирование алгоритма. Практическая работа	124
5.5–5.6. Тестирование программы.....	128
5.7. Практические задания. Подготовка к олимпиаде	134
Словарь	137
Глоссарий	144
Использованная литература.....	149

Оқулық басылымы Учебное издание

Кадиркулов Роман Алауович
Рыскулбекова Асима Даулетбековна
Нурмуханбетова Гулира Кенжеевна

ИНФОРМАТИКА ИНФОРМАТИКА

Жалпы білім беретін мектептің 8-сынып оқушыларына арналған оқулық
 Учебник для учащихся 8 класса общеобразовательной школы

Редакторлары / Редакторы – *М.Г. Важенина, А.А. Сапарғалиева*

Мұқаба / Обложка – *Е. Жузбаев*

Беттеуші / Верстка – *Г. Тумарбайқызы*

Басуға 09.07.2021 ж. қол қойылды. Подписано в печать 09.07.2021 г.
 Пішімі 70x100 ¹/₁₆, Есептік баспа табағы 5,87. Формат 70x100 ¹/₁₆. Уч.-изд.л. 5,87.
 Шартты баспа табағы 12,35. Офсеттік басылым. Усл.печ.л. 12,35. Печать офсетная.
 Әріп түрі «DS SchoolBook». Офсеттік қағаз. Гарнитура «DS SchoolBooks». Бумага офсетная.
 Таралымы 48 000 дана. Тапсырыс № 3090. Тираж 48 000 экз. Заказ № 3090

Сапасы жөнінде шағымыңыз болса, С претензиями по качеству обращаться:
 мына мекемеге хабарласыңыз: Республика Казахстан,
 «АЛМАТЫКІТАП БАСПАСЫ» ЖШС, ТОО «АЛМАТЫКІТАП БАСПАСЫ»
 050012, Алматы қаласы, Жамбыл көшесі, 111-үй, 050012, г. Алматы, ул. Жамбыла, 111,
 тел. (727) 250 29 58, факс: (727) 292 81 10. тел. (727) 250 29 58; факс: (727) 292 81 10.
 e-mail: info@almatykitap.kz e-mail: info@almatykitap.kz

Сапа және қауіпсіздіктің барлық стандартына сай. Соответствует всем стандартам качества и безопасности.
 Сертификаттау қарастырылмаған. Сертификация не предусмотрена.
 Сақтау мерзімі шектелмеген. Срок годности не ограничен.

Түркияда басылды / Отпечатано в Турции

Avea Basım Yayın San. ve Tic. Ltd.Sti

Cihangir mah. Guvercin cad. No:3/1. Baha is merkezi A Blok Kat:2 34310. Haramidere - Istanbul

Приобрести книги можно в книжных магазинах ТОО «АЛМАТЫКІТАП БАСПАСЫ»

г. Нур-Султан: ул. Иманова, 10, тел.: (7172) 53 70 84, 27 29 54;
 пр. Б. Момышулы, 14, тел.: (7172) 42 42 32, 57 63 92; пр. Жеңіс, 67, тел.: (7172) 29 93 81; 29 02 12.
 г. Алматы: пр. Абая, 35/37, тел.: (727) 267 13 95, 267 14 86;
 ул. Гоголя, 108, тел.: (727) 279 29 13, 279 27 86; ул. Кабанбай батыра, 109, тел.: (727) 267 54 64, 272 05 66;
 ул. Жандосова, 57, тел.: (727) 303 72 33, 374 98 59; пр. Гагарина, 76, тел. (727) 338 50 52;
 ул. Майлина, 224 «А», тел. (727) 386 15 19; ул. Толе би, 40/1, тел.: (727) 273 51 38, 224 39 37.

Интернет-магазин www.flip.kz

Коммерческий отдел, тел.: (727) 292 92 23, 292 57 20.

e-mail: sale1@almatykitap.kz

Об имеющихся книгах и новинках
 вы можете узнать на сайте www.almatykitap.kz

Оқулықты пайдалану туралы деректер. Сведения о пользовании учебником

№	Оқушының аты-жөні Фамилия и имя ученика	Оқу жылы Учебный год	Оқулықтың жағдайы Состояние учебника	
			Жылдың басында В начале года	Жылдың аяғында В конце года
1				
2				
3				
4				
5				