



## Урок 2. Простые и составные условия. Оператор ветвления

Задачи урока: Содействовать формированию умений записывать простые и составные условия, алгоритмическую конструкцию «ветвление».

Создавать условия для воспитания самостоятельности, коллективизма, ответственности за себя и других членов коллектива;

Способствовать развитию мышления, умения применять полученные знания при

Учащиеся должны знать: алгоритмическую конструкцию «ветвление»



### Оборудование:

- компьютеры (смартфоны, планшеты)
- доступ в Интернет

### Интернет-ресурсы:

- [Сайт учреждения образования](#) Обучение/Информатика
- [Google Диск 8\\_2 Простые и составные условия. Оператор ветвления](#)

### ТЕХНОЛОГИЧЕСКАЯ КАРТА УРОКА

Этап урока	Учитель	Ученик
1. Организационный момент.	Вход в урок. Создание условий для положительной мотивации учащихся. Рефлексия_ Начало урока	
2. Актуализация знаний	<a href="#">Тест Повторение основ алгоритмизации и программирования</a>	
3. Изучение нового материала	Простые и составные условия. Составной оператор. Оператор ветвления	
4. Закрепление знаний	 § 1 (? 1-5), стр. 16. Анализ программ: пример 1-6	
	 ПР: упр. 2-6	

5. Контроль знаний	Тест Оператор ветвления
6. Итог урока. Рефлексия	<a href="#">Рефлексия_Облако тегов</a>
7. Домашнее задание	§ 1, упр. 1.

### Ход урока

Проводим мы урок в необычной форме – в форме игры.

*Эпиграф:*

*«Задача, конечно, не очень простая  
Играя – учить, и учиться – играя.  
Но если с учебой связать развлечение,  
То праздником станет любое ученье!»*

Я надеюсь, что сегодняшнее наше с вами общение станет для вас действительно праздником.

Вы, наверное, хотите меня спросить, а во что же мы будем сегодня играть? А играть мы будем в сыщиков и проведем настоящее расследование. Сегодня в процессе игры вам понадобятся все ваши знания по теме, навыки работы на компьютере.

После выполнения каждого очередного задания вы будете получать “след” — ключ к следующему заданию. Всего заданий 4. Ответы должны быть обязательно записаны в тетрадях.

Я — начальник следственного отдела. Вы — следственные группы. Кроме того есть Группа экспертов. Учитель: Каждая группа к концу игры должна выявить существенные данные по этим вопросам. За каждое правильно выполненное задание вы будете получать 5 баллов (Слайд 2) К нам поступило заявление с вопросами: К нам поступило заявление с вопросами:

1) Когда была создана первая ЭВМ и как она называлась?  
2) Под чьим руководством были изобретены - первый персональный компьютер **Macintosh**, использующий графический интерфейс, плеер iPod и мобильный телефон iPhone?

(Слайд 3) Баллы – это ваша оценка. В конце игры мы подсчитаем общую оценку на группу. Начнем с разминки.

#### Этап 1. “О чем разговор?”

Учитель: Вам предлагается по заданным наборам карточек - собрать домино терминов. Недействующие слова свяжите с темой занятия.

(Слова на последней карточке с другой стороны - Составление программ на языке программирования Pascal)

Учащиеся в группах обсуждают задание, записывают ответ. По окончании работы капитан зачитывает задание и ответ. Учитель подводит итоги: расставляет в таблице места.

Хороший следователь должен уметь проверять достоверность фактов, получаемых во время расследования, а как вы с этим справитесь, мы сейчас посмотрим. В предложенных программах найдите ошибки:

(1 человек от группы – остальные собирают термины)

1 группа	2 группа	3 группа	4 группа
<pre> Program R2; var a integer; begin writeln(Введите число'); readln(a); if a mod 2 = 0 then writeln('а-четное' ) else writeln('а-нечетно е'); readln end.</pre>	<pre> Program ; Var b,s: real Begin Writeln (Введите числа b и c') Readln( b,c); A=b+c; Write ln ( 'a=',a); End.</pre>	<pre> Program ; Ver a,b,c,x,y: real; Begin Writeln ('Введите числа a, b и c') Readln( a,b,c); D:=sqr(b)-4*a*c; x:=(-b+sqrt(d))-2 *a; Writeln ( 'x=',x); And.</pre>	<pre> program R1; var a,b,c,d,n: integer; begin writeln('Введите 4 числа') readln(a,b,c,d); m=a*d; n:=b*c; writeln('числитель=', m); writeln('знаменатель=' , n); read n end.</pre>

Учитель: Итак, переходим ко 2-му этапу.

(Слайд 4) **Этап 2. “Сбор фактов”.**

Учитель: Вы должны из приведенных блок-схем “добыть факты” — данные из биографии ученого, который является изобретателем первой цифровой электронной вычислительной машины. Данные получатся в том случае, если группы будут работать оперативно и слаженно!

Капитаны групп получают карточки-задания.

(Слайд 5) **1 Команда**

1 апреля 1976 г. вместе со Стивом Возняком основал компанию Apple. ПК Apple II на протяжении 5 был самым популярным компьютером.

(Слайд 6) **2 Команда**

24 января 1984 года представил Macintosh- первый коммерчески успешный маленький компьютер с графическим интерфейсом пользователя.

(Слайд 7) **3 Команда**

Стал одним из основателей анимационной студии Pixar. Под его руководством Pixar выпустила такие фильмы, как "История игрушек" и "Корпорация монстров" и другие. Вернулся в компанию Apple в 1996 году.

(Слайд 8) **4 Команда**

В 2001 году представил первый плеер iPod. В 2007 году сенсорный мобильный телефон iPhone, который своим мобильным браузером произвёл революцию в мобильном вебе.

(Учащиеся в группах, «выполняя» блок-схему, записывают сведения. Капитаны их зачитывают. Группа экспертов заполняет таблицу результатов)

Учитель: Каким образом связано задание с темой занятия?

Учащиеся: Нужно уметь не только составлять блок-схему, но и “читать” её и понимать.

(Слайд 9) **Этап 3. “Определи номер”.**

Учитель: Часто случается, что важную информацию помещают не туда, где ей следует быть. Известно лишь, что задания находятся в ячейках сейфа, номера вы сможете определить,

выполнив условия: вы должны найти числа, которые являются результатами выполнения предоставленных вам фрагментов программ. Полученное число и есть номер нужной ячейки камеры хранения.

*(Слайд 10)* **1 Команда**

```
a:=9; b:=1;c:=12;  
if (a<=b) or (a>c) then x:=5*a-b  
    else x:=3*a+b;  
    end.
```

*(Слайд 11)* **2 Команда**

```
a:=5; b:=4;c:=3;  
if (a<b) and (a>c) then x:=4*a-b*a  
    else x:=a-b;  
    end.
```

*(Слайд 12)* **3 Команда**

```
a:=-2; b:=7;c:=3;  
if (a<b) and (b>c) then x:=3*a+b+c  
    else begin  
        a:=b+c;  
        x:=a-3*b;  
    end.
```

*(Слайд 13)* **4 Команда**

```
a:=15; b:=9;c:=11;  
if (a>=b) or (a<c) then x:=a+b  
    else x:=2*a-4*b;  
    end.
```

Капитаны называют номер ячейки. Если ответ верный, то экспертная группа фиксирует результат в таблице и выдаёт следующее задание – Этап №4. В противном случае команда совместно ищет ошибку и устраняет её или прибегает к помощи учителя)

Учитель: Что нам дает выполнение подобного задания?

Учащиеся: Нужно уметь не только составлять программу, но и “читать” ее, понимать, как работает каждый оператор.

*(Слайд 16) Этап 4. “Внимание!”*

Учитель: Вы успешно справились с предыдущим заданием, и теперь каждая группа получила своё задание на розыск:

установить фамилию и дату рождения изобретателя **Macintosh**?

отыскать фотографию этого человека;

определить, где родился этот человек;

Определить, когда была создана первая ЭВМ.

Учащиеся групп получают задания:

Каждый группа выбирает и решает разноуровневые задачи.

Критерии оценки:

Задания 1 уровня сложности «Составить блок схему к предложенному алгоритму» – 3 балла

Задания 2 уровня сложности «Составить программу по блок-схеме»– 4 балла

Задания 3 уровня сложности «Самостоятельно составить программу» – 5 баллов

*(Слайд 17) 1.* Вам нужно определить человека, под руководством которого в 2007 г. был изобретен мобильный телефон iPhone, если известно, что в то время он был не старше 55.

**Стивен Пол Джобс (1955 -2011 )**

**Стивен Возняк (1950- н.в.)**

**Деннис Ритчи (1941-2011)**

(составить программу)

*(Слайд 19) 2.* Вам необходимо по фотографии узнать Стива Джобса, если известно, что он темноволосый мужчина с очками.

(поиск оформить в виде блок-схемы)

*(Слайд 20) 3.* Определить, когда была создана первая ЭВМ, если известно, что в этом году будет 65 лет со дня этой замечательной даты.

Варианты ответа:

1) 1949 г;

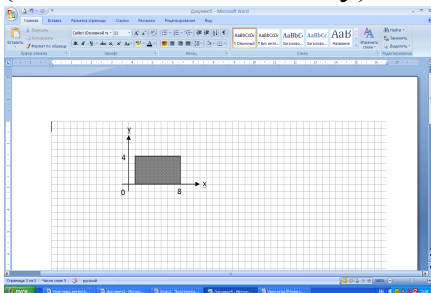
2) 1946 г;

3) 1945г;

4) 1948 г.

*(Слайд 22) 4.* Определить район поиска (город, в котором родился Стив Джобс, если известно, что поиск необходимо осуществлять в пределах следующей фигуры на карте мира:

(составить блок-схему)



Города, имеют координаты:

(2; 5) —

(7; 3) —

(6; 5) —

Ответ:

(Учащиеся в группах обсуждают задание, составляют алгоритмы и программы, отлаживают программы на компьютерах и получают ответы. Учитель в ходе отчета задает вопросы по составлению алгоритма и написанию программы. Экспертная группа подводит итоги этого этапа, распределяет места)

Учитель: Чем интересно было это задание?

Учащиеся: Отрабатывали навыки в составлении алгоритма и программы; отладки программы; необычность формы записи условия задачи.

Учитель выставляет в журнал среднюю оценку за командную игру и индивидуальную работу.

Итак, оценки за урок ...

Учитель: Ответьте на вопросы

- Что мне удалось на сегодняшнем уроке-игре?
- Что нового узнали?
- Мои ошибки?
- Какие причины стоят за этим?
- Что мне требуется понять, осмыслить, чтобы не повторять эти же ошибки на следующих уроках?

Оцените уровень своих знаний и умений по этой теме на данный момент.

(Учащиеся на лестнице знаний ставят своих роботов)

*(Слайд 23) Мне хотелось бы закончить наш урок словами С. Джобса:*

*Если ты еще не нашел того, что искал, продолжай поиски. Не останавливайся. Поскольку это касается сути вещей, ты не пропустишь момента, когда действительно найдешь это.*

### **Домашнее задание**

Приложение.

1. Идея урока, структура  
<http://nsportal.ru/shkola/informatika-i-ikt/library/library/obobshchayushchii-urok-po-teme-razvetvlyayushchiesya-algori>
2. Материалы  
<http://ru.wikipedia.org>

**Программа Б.**  
**Мы – дети Галактики ...**

км).

Пусть D1, D2 – диаметры двух планет, тогда возможны три случая:

- 1) первая планета имеет больший диаметр, если  $D1 > D2$ ;
- 2) первая планета имеет меньший диаметр, если  $D1 < D2$ ;
- 3) диаметры планет одинаковы, если  $D1 = D2$ .

**If P Then S1 Else S2;**

**ВЕТВЛЕНИЕ  
(полное)**

**Program Kosmos;**

**Var D1, D2 : Real ;**

**Begin**

**Write ('Диаметр Урана= ');**

**Readln (D1) ;**

**Write ('Диаметр Нептуна= ');**

**Readln (D2) ;**

**Writeln ('У какой планеты больший диаметр? ');**

**If D1>D2 Then**

**Writeln ('у Урана')**

**Else If D1<D2 Then**

**Writeln ('у Нептуна ')**

**Else**

**Writeln ('Диаметры обеих планет равные ')**

**End.**

Если условие P истинно, то выполняется оператор S1 и затем оператор, следующий за символом «;». Если условие P ложно, то выполняется оператор S2 и затем оператор, записанный после «;».

### Алгоритмы с ветвлением. Оператор ветвления IF

Ветвление – это такая форма организации действий, когда при проверке условия выполняется одна либо другая последовательность действий

#### Краткая форма ветвления

- 1) **If <условие> then <оператор>;**

2) If <условие> then begin  
 <оператор 1>;  
 <оператор 2>;  
 ...  
 <оператор n>;  
 end;

Если условие ИСТИНО выполняется оператор или группа операторов после THEN.

*Блок-схема*

Полная форма ветвления

3) If <условие> then <оператор 1> else <оператор 2> ;

Если условие ИСТИНО выполняется оператор 1 или группа операторов после THEN, если ЛОЖЬ - оператор 2 или группа операторов после ELSE

*Блок-схема*

Если после служебных слов THEN или ELSE стоит группа операторов то она заключается в операторские скобки begin ... end

Задача 1. Даны два числа a и b. Написать программу для нахождения их частного	
<pre> Program Z1; Var a,b,c: real; Begin Writeln ('введите числа a и b:'); Read(a,b); c:=a/b; Writeln('c=',c); End. </pre>	<pre> Program Z1; Var a,b,c: real; Begin Writeln ('введите числа a и b:'); Read(a,b);  End. </pre>

Задача 1 Определите является ли число x - целое.

Задача 2 Определите является ли целое число x кратным целому числу a.

Задача 3 Даны три числа a,b,c. Возвести их в квадрат, если a>0, b>0, c>0. В противном случае возвести в куб

Алгоритмы с ветвлением. Оператор ветвления IF

Ветвление – это такая форма организации действий, когда при проверки условия выполняется одна либо другая последовательность действий

Краткая форма ветвления

1) If <условие> then <оператор>;

2) If <условие> then begin  
 <оператор 1>;  
 <оператор 2>;  
 ...  
 <оператор n>;  
 end;

Если условие ИСТИНО выполняется оператор или группа операторов после THEN.

*Блок-схема*

Полная форма ветвления

3) If <условие> then <оператор 1> else <оператор 2> ;

Если условие ИСТИНО выполняется оператор 1 или группа операторов после THEN, если ЛОЖЬ - оператор 2 или группа операторов после ELSE

*Блок-схема*

Если после служебных слов THEN или ELSE стоит группа операторов то она заключается в операторские скобки begin ... end

Задача 1. Даны два числа a и b. Написать программу для нахождения их частного
---

<pre> Program Z1; Var a,b,c: real; Begin Writeln ('введите числа a и b:'); Read(a,b); c:=a/b; Writeln('c=',c); End.</pre>	<pre> Program Z1; Var a,b,c: real; Begin Writeln ('введите числа a и b:'); Read(a,b);  End.</pre>
---	---

**Задача 1** Определите является ли число  $x$  - целое.

**Задача 2** Определите является ли целое число  $x$  кратным целому числу  $a$ .

**Задача 3** Даны три числа  $a, b, c$ . Возвести их в квадрат, если  $a > 0, b > 0, c > 0$ . В противном случае возвести в куб



1. Запустите программу и проверьте правильность ее работы.
2. Измените программу для работы с другими планетами Солнечной системы (требуемые данные заимствуйте из справочной литературы).
3. Измените программу для работы с другими характеристиками планет Солнечной системы (например, количеством природных спутников; так, у Нептуна их два, а у Урана – пять).
4. Известно, что форма орбиты движения небесных тел зависит от их масс, расстояния между ними и от скорости, с которой одно тело движется относительно другого. Если тело массой  $m_1$  (кг) находится на расстоянии  $r$  (м) от тела массой  $m_0$  (кг) и движется в этот момент со скоростью  $v$  (м/с), то форма орбиты определяется величиной  $h = v^2 - 2(m_0 + m_1)f / r$ .  
 Постоянная тяготения  $f = 6,673 \cdot 10^{-11} \text{ м}^3 / \text{кг} \cdot \text{с}^2$ . Если  $h < 0$ , то тело  $m_1$  движется относительно тела  $m_0$  по эллиптической орбите; если  $h = 0$  – по параболической орбите; если  $h > 0$  – по гиперболической орбите. Напишите программу вывода формы орбиты движения тела (кометы, спутника Земли) относительно планеты Земля.
5. Напишите программу нахождения наибольшего значения роста у двух членов школьной баскетбольной команды (рост спортсменов вводится с клавиатуры).

Проект урока (с использованием интерактивной доски)

**Класс:** 8

**Тема урока:** Алгоритмы с ветвлением. Оператор ветвления IF

**Тип урока:** изучение нового материала

**Форма проведения:** интегральная технология

**Цель урока (когнитивная):** предполагается, что к окончанию урока ученики будут:

- понимать работу оператора ветвления IF;
- применять оператор ветвления IF при решении задач.

**Задачи личностного развития учащихся:**

- создать условия для развития логического мышления, коммуникативных качеств учащихся, умений высказывать гипотезы, формулировать выводы;
  - создать условия для воспитания культуры умственного труда и информационной культуры.
- Оборудование:** интерактивная доска, компьютеры учеников, ориентировочные карты по теме урока для каждого учащегося.

## Сценарий урока

### 1. Организационно – мотивационный этап (6 мин).

Вопросы

1. Какие типы алгоритмов вы знаете.
2. Что такое алгоритм с ветвлением.

А сейчас решим задачу.

**Задача 1.** Даны два числа *a* и *b*. Написать программу для нахождения их частного (величины *c*).

**слайд 1**

*Program Z1;*

*Var*

*a,b,c: real;*

*Begin*

*Writeln ('введите числа a и b:');*

*Read(a,b);*

*c:=a/b;*

*Writeln('c=',c)*

*End.*

Предложить ученикам по очереди произнести по одной команде программы.

Когда программа составлена продолжить опрос

### 3. Когда данная программа не сможет работать (выдаст сообщение об ошибке)?

(когда *b* будет равняться 0)

А можно ли исправить программу так, чтобы при вводе  $b=0$  выдавался ответ, что деление нуль невозможно. Конечно да, но для решения таких задач необходимо знать оператор ветвления.

Итак, тема сегодняшнего урока «оператор ветвления IF». (Раздать опорные конспекты) **слайд 2**

### 2. Операционно – познавательный этап (21 мин).

Напоминаю **Ветвление** – это такая форма организации действий, когда при проверки условия выполняется одна либо другая последовательность действий.

Демонстрирую слайд 2 с комментариями.

Существуют 2 формы ветвления

#### Краткая форма ветвления **слайд 3**

1) If <условие> then <оператор>;

2) If <условие> then begin

<оператор 1>;

<оператор 2>;

...

<оператор n>;

end;

<p>Если условие ИСТИНО выполняется оператор или группа операторов после THEN</p>
--

Можно привести пример из быта: Если ты сделаешь домашнюю работу, то пойдёшь гулять

*Блок-схема*

#### Полная форма ветвления **слайд 4**

3) If <условие> then <оператор 1> else <оператор 2>;

<p>Если условие ИСТИНО выполняется оператор 1 или группа операторов после THEN, если ЛОЖЬ - оператор 2 или группа операторов после ELSE</p>
---

*Блок-схема*

Если ты сделаешь домашнюю работу, то пойдёшь гулять, иначе будешь сидеть дома.

Обратить ещё раз внимание на то что: *Если после служебных слов THEN или ELSE стоит группа операторов то она заключается в операторские скобки begin ... end. Такой оператор называется составным*

Для закрепления рассматриваем несколько примеров с оператором IF. Предлагаю обсудить в парах решения данных примеров.

**слайд 5**

**Пример 1** Чему будет равно значение S и P после выполнения фрагмента программы

S:=10; P:=1;

If S > 3 then S:=s+2\*P;

**Условие истинно**

**Значит выполнится оператор после then**

**S=12**

**слайд 6**

**Пример 2** Чему будет равно значение S и P после выполнения фрагмента программы

S:=1; P:=1;

If S > =P then begin S:=S+P; P:=S-P; end;

**Условие истинно**

**Значит выполнится группа операторов после then**

**S=2 P=1**

**слайд 7**

**Пример 3** Чему будет равно значение S и P после выполнения фрагмента программы

S:=5; P:=1;

If S < =P then begin S:=S / P; P:=S\*P; end;

**Условие ложно**

**S и P останутся прежними**

**S=5 P=1**

**слайд 8**

**Пример 4** Чему будет равно значение S и P после выполнения фрагмента программы

S:=10; P:=1;

If S > =P then begin S:=S+P; P:=S-P; end

Else begin S:=10+P; P:=10-S; end;

**Условие истинно**

**Значит выполнится группа операторов после then**

**S=11 P=9**

**слайд 9**

**Пример 5** Чему будет равно значение S и P после выполнения фрагмента программы

S:=10; P:=1;

If S < P then begin S:=S+P; P:=S-P; end

Else S:=10+P;

**Условие ложно**

**Значит выполнится оператор после else**

**S=11 P=1**

**слайд 10**

**Пример 6** Чему будет равно значение S и P после выполнения фрагмента программы

S:=1; P:=1;

If S < =P then begin S:=S+P; P:=S-P; S:=S\*P; end

Else S:=10+P;

**Условие истинно**

**Значит выполнится группа операторов после then**

**S=2 P=1**

**слайд 11**

**Пример 7** Чему будет равно значение S и P после выполнения фрагмента программы

S:=1; P:=1;

If S < P then begin S:=S+P; P:=S-P; S:=S\*P; end

else if S>P then S:=10+P

else begin S:=S+1; P:=S-1; S:=S\*P; P:=S; end;

**1 условие ложно проверяется 2 условие, оно также ложно**

**Значит выполнится группа операторов после else**

**S=0 P=0**

А сейчас вернёмся к нашей задаче **слайд 12**

**Задача 1.** Даны два числа a и b. Написать программу для нахождения их частного (величины c).

Предлагаю ученикам подумать и исправить программу, чтобы она не выдавала ошибку.

**Program Z1;**

**Var**

**a,b,c: real;**

**Begin**

**Writeln ('введите числа a и b:');**

**Read(a,b);**

**c:=a/b;**

**Writeln('c=',c);**

**End.**

**Program Z1;**

**Var**

**a,b,c: real;**

**Begin**

**Writeln ('введите числа a и b:');**

**Read(a,b);**

**If b=0 then writeln ('делить на 0 нельзя') else**

**begin c:=a/b; Writeln('c=',c); end;**

**End.**

## **ФИЗКУЛЬТМИНИТКА (на выбор учителя)**

### **3. Контрольно – коррекционный этап (15 мин).**

#### **слайд 13**

На этом этапе ученикам предлагается на выбор 1 из 3 задач. Перед самостоятельной работой разбираем все 3 задачи.

Далее проводится самопроверка и самоанализ результатов самостоятельной деятельности.

### **4.Рефлексивно – диагностирующий этап (2 мин).**

- Что на уроке вы узнали для себя нового?
- Когда используется оператор ветвления IF?
- Какие формы ветвления Вы знаете?
- В чём основное отличие краткой формы ветвления от полной?
- Когда используется операторские скобки begin ... end?
- Что будет если опустить операторские скобки после служебного слова THEN в полной форме ветвления?
- А как вы думаете, что мы будем делать на следующем уроке?

### **5. Организация домашней работы (1 мин.)**

Опорный конспект. Оформить все задачи из опорного конспекта в тетради

## **Тема12: «Основы алгоритмизации и программирования»**

### **Урок 2: Алгоритмическая конструкция ветвление. Запись на языке программирования**

#### **Цели урока:**

- закрепить теоретические знания по теме путем выполнения входного задания;
- ввести понятие составного оператора.
- научить правильно применять структуру ветвления при решении различных видов задач по теме;
- развивать интерес к учению, самостоятельности, стремления к расширению кругозора;
- развивать умения анализировать, сравнивать, строить аналогии, обобщать и систематизировать.
- воспитывать дисциплинированность, любознательность, терпение.

#### **План урока:**

- I. Организационно-мотивирующий этап.
- II. Повторение + проверка знаний по пройденному материалу.
- III. Объяснение нового материала
- IV. Закрепление нового материала при решении задач.
- V. Контрольно-оценочный этап.
- VI. Рефлексия.
- VII. Д/з.



## Ход урока.

I. Организационно-мотивирующий этап. Объявление темы и целей урока.

II. Проверка + Повторение пройденного материала.

Учащиеся загружают файл **СОСТАВНЫЕ УСЛОВИЯ** и выполняют задания.

На доске нарисованы блок-схемы.

Пример 1

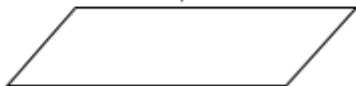
Начало



Пример 1

Заполните блок-схему для решения следующей задачи.

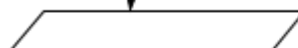
Дано целое число  $n$ . Если оно положительное, то вычезть из него 1, в противном случае прибавить к нему 10.



Конец

Пример 2

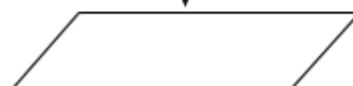
Начало



Пример 2

Заполните блок-схему для решения следующей задачи

Дано целое число  $m$ . Если число нечетное, то прибавить к нему 1.



Конец

Два ученика получают задания на карточках.

После

выполнения задания идет проверка и анализ заданий, выполненных на доске

### III. Объяснение нового материала

Для записи команды ветвления на языке программирования Pascal используется условный оператор языка **If**

**If** <условие> **then** <оператор\_1> **else** <оператор\_2>;                    **полная форма**  
**If** <условие> **then** <оператор\_1>;    **сокращенная форма**

Рассмотрим на примерах, как работает оператор if, используя блок-схемы, выполненные на доске.

<pre>program primer1;   var n: real; begin   write('Введите число '); readln(n);   if n &gt; 0 then n:=n-1 else n:=n+10;   writeln(n); end.</pre>	<pre>program primer2;   var m: integer; begin   write('Введите число '); readln(m);   if m mod 2 = 1 then m:=m+1;   writeln(m); end.</pre>
---	--

Для записи составных условий на языке программирования используют следующие логические операции:

- and** – логическое «и»;
- or** – логическое «или»;
- not** – логическое отрицание.

Простые условия при этом обязательно заключаются в скобки, так как логические операции имеют более высокий приоритет, чем операции сравнения.

*Например:*

- (2>5) and (2>3) - неверно (ложно)
- (3>2) or (3<1) - верно
- not (5 > 6) - верно

#### Составной оператор

При составлении программ на языке программирования часто бывает так, что в случае выполнения либо невыполнения некоторого условия в операторе if необходимо осуществить несколько действий. В этом случае *последовательность действий (несколько операторов подряд) объединяют в одну группу, заключенную между словами begin и end.*

*Общий вид составного оператора может выглядеть следующим образом:*

```
Begin
Оператор1;
Оператор2;
.....
End;
```

Зарезервированные слова **Begin** и **End** часто называют открывающей и закрывающей **операторными скобками**.

Рассмотрим пример, демонстрирующий использование составного оператора.

**Пример 3.** Пусть дано некоторое целое число. Требуется составить программу, которая выводит на экран монитора сообщение о том, какое это число: положительное либо отрицательное. И если это число положительное, увеличить его в 2 раза, в противном случае возводит его в квадрат.

Алгоритм решения данной задачи в виде блок-схемы.

Программа, реализующая данный алгоритм, может выглядеть так:



```

program primer3;
  var a,r: integer;
begin
  write('Введите целое число: ');
  readln(a);
  if a > 0 then
    begin writeln('положительное');
      r:=a * 2;
    end
  else
    begin writeln('отрицательное');
      r:=a*a;
    end;
  write ('r=',r);
end.

```

Результат выполнения программы:

а	Ответ:
-5	отрицательное r =25
8	положительное r =16

# Закрепление нового материала при решении задач

## Упражнения

1. Даны два числа  $a, b$  ( $a, b \neq 0$ ). Ответить одного ли знака введенные числа.

a	b	Ответ:
$a > 0$	$b > 0$	Одного знака
$a < 0$	$b < 0$	Одного знака
$a < 0$	$b > 0$	Разные знаки
$a > 0$	$b < 0$	Разные знаки

```
program ypr1;  
  var a,b: real;  
begin  
  write('Введите два числа: ');  
  readln(a,b);  
  if (a > 0) and (b > 0) then writeln(' Одного знака ');  
  if (a < 0) and (b < 0) then writeln(' Одного знака ');  
  if (a < 0) and (b > 0) then writeln(' Разные знаки ');  
  if (a > 0) and (b < 0) then writeln(' Разные знаки ');  
end.
```

Результат выполнения программы

a	b	Ответ:
$5 > 0$	$9 > 0$	Одного знака
$-7 < 0$	$-12 < 0$	Одного знака
$-6 < 0$	$8 > 0$	Разные знаки
$8 > 0$	$-4 < 0$	Разные знаки

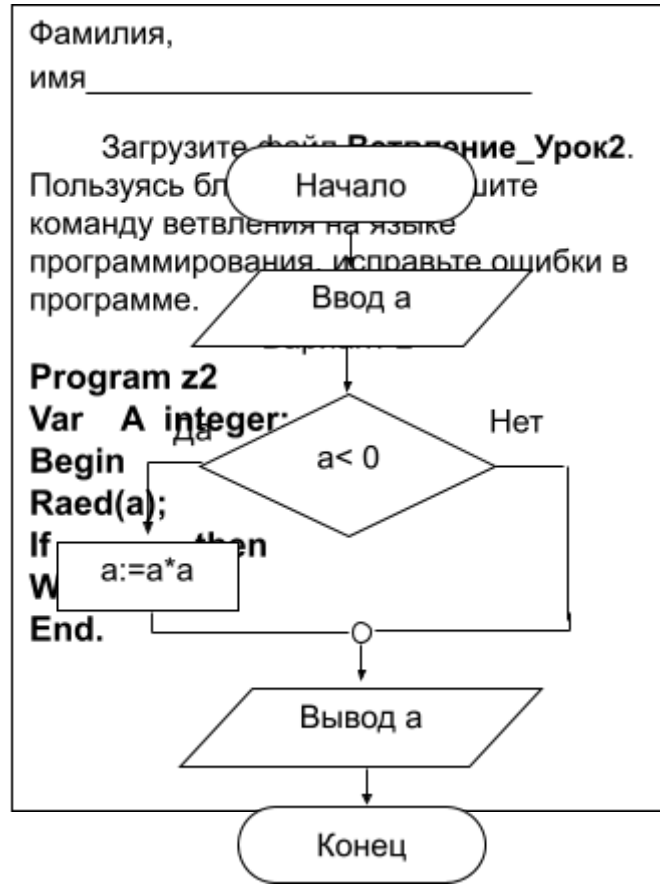
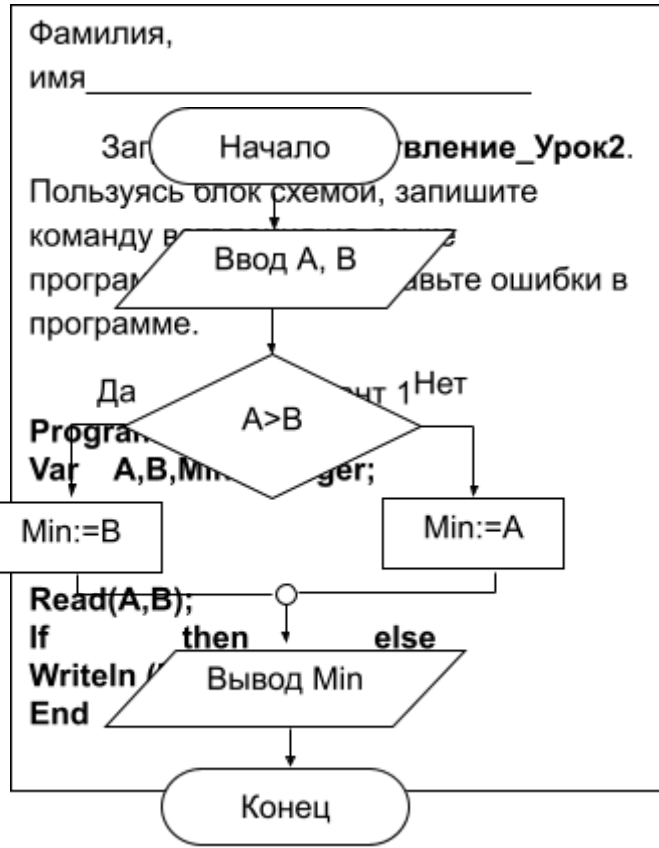
2. Дано целое число. Написать программу, определяющую является ли введенное число двузначным. Если число двузначное найти сумму цифр этого числа.

```
Program ypr2;  
  var a, c1,c2,C:integer;  
begin  
  write ('Введите целое число: '); readln (a);  
  if (a>=10) and (a<=99) then  
    begin  
      writeln ('число двузначное');  
      c2:=a mod 10;  
      c1:=a div 10;  
      C:=c1+c2;  
      write('сумма цифр =',C);  
    end  
  else write('число не двузначное');  
end.
```

	a	результат
1	8	число не двузначное
2	18	число двузначное сумма цифр=9
3	145	число не двузначное

## VI. Контрольно-оценочный этап

Учащиеся получают карточки задания



Задания  
в файле:

## VII. Д/з.:

Задача 1. Написать программу, определяющую, является ли человек возраста N – школьником? (школьник в возрасте от 6 до 18 лет).

```
program zadaca1;  
  var N: integer;  
begin  
  write('Введите возраст человека в годах '); readln(N);  
  if (n >=6) and (n <=10) then writeln ('школьник')  
  else writeln (' не школьник');  
end.
```

Задача 2. Дана точка с ненулевыми координатами (x;y). Написать программу, определяющую в какой координатной плоскости находится точка.

```
Program zadaca2;  
var x,y:integer;  
begin  
  write ('Введите координаты точки '); readln (x,y);  
  if (x>0) and (y>0)then write ('точка находится в 1 четверти');  
  if (x<0) and (y>0)then write ('точка находится в 2 четверти');  
  if (x<0) and (y<0)then write ('точка находится в 3 четверти');  
  if (x>0) and (y<0)then write ('точка находится в 4 четверти');  
end.
```

	x	y	результат
1	8	5	точка находится в 1 четверти
2	-8	5	точка находится в 2 четверти
3	-8	-5	точка находится в 3 четверти
4	8	-5	точка находится в 4 четверти

Задача 3. Даны три стороны треугольника a, b, c. Написать программу, определяющую площадь треугольника, если треугольник существует. (треугольник существует если для его сторон выполняется условие  $|b-c| < a < b+c$ ).

```
program zadaca3;  
  var a,b,c,S: real;  
begin  
  write('Введите стороны треугольника ');  
  readln(a,b,c);  
  if (a < b+c ) and (a > abs(b-c)) then  
    begin writeln(' треугольник существует ');  
    p:=(a+b+c)/2;  
    S:=sqrt(p*(p-a)*(p-b)*(p-c));  
    write('площадь треугольника =',S);  
    end  
  else writeln(' треугольник не существует ');  
end.
```

	a	b	c	результат
1	4	3	5	треугольник существует площадь треугольника =6
2	2	5	7	треугольник не существует