

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ  
ДВОРЕЦ ДЕТСКОГО (ЮНОШЕСКОГО) ТВОРЧЕСТВА ВЫБОРГСКОГО РАЙОНА  
САНКТ-ПЕТЕРБУРГА  
(ГБУ ДО ДДОТ)

Практические занятия  
Модуль «Программирование на языке Pascal»

**«Основы компьютерных технологий»**

Второй год обучения  
Группа № 1  
Возраст учащихся 9 — 12 лет

**Ухтина Елена Николаевна,**  
педагог дополнительного образования

Санкт-Петербург  
**2016 / 2017 учебный год**

Цель занятий: познакомить учащихся с алгоритмом с условием (ветвление)

**Задачи:**

Ⓢ *Обучающие:*

Ⓢ сформировать у учащихся первичные знания по применению изученного материала.

Ⓢ *Развивающие:*

Ⓢ учить анализировать, обобщать и систематизировать;

Ⓢ обогащать словарный запас учащихся.

Ⓢ *Воспитательные:*

Ⓢ развивать информационную культуру учащихся, способность к самостоятельной и коллективной деятельности, рефлексия.

**Тип занятия:** изучения и первичного закрепления новых знаний.

**Оборудование:** мультимедийный проектор, компьютеры с установленной средой программирования Pascal, настроенная локальная сеть, конспект для учащегося раздаточные карточки-задания.

## Занятие 1. Алгоритмы с условием (Ветвления). Найти максимальное число из трех

### Задача

Вводятся три целых числа. Определить какое из них наибольшее.

### Пояснение к задаче и алгоритм решения

Пусть  $a$ ,  $b$ ,  $c$  - переменные, которым присваиваются введенные числа, а переменная  $m$  в конечном итоге должна будет содержать значение наибольшей переменной. Тогда алгоритм программы сведется к следующему:

1. Сначала предположим, что переменная  $a$  содержит наибольшее значение. Присвоим его переменной  $m$ .
2. Если текущее значение  $m$  меньше, чем у  $b$ , то следует присвоить  $m$  значение  $b$ . Если это не так, то не изменять значение  $m$ .
3. Если текущее значение  $m$  меньше, чем у  $c$ , то присвоить  $m$  значение  $c$ . Иначе ничего не делать.

Данную задачу можно решить и без использования четвертой переменной, однако программа будет логически сложнее:

1. Если  $a$  больше  $b$ , то проверить больше ли  $a$ , чем  $c$ . Если это так, то максимальное число содержит переменная  $a$ . Если  $a$  больше  $b$ , но при этом меньше  $c$ , то максимальным является значение  $c$ .
2. Иначе (когда  $a$  меньше  $b$ ) сравнить значения  $b$  и  $c$ . Большее из них и есть максимальное.

### Исходный код на языке программирования Pascal

// 1-й способ:

```
var
a,b,c,max: integer;
begin
readln(a,b,c);
max := a;
if max < b then max := b;
if max < c then max := c;
writeln(max);
end.
```

// 2-й способ:

```
var
a,b,c: integer;
begin
readln(a,b,c);
if a > b then
if a > c then writeln(a)
else writeln(c)
else
if b > c then writeln(b)
else writeln(c);
end.
```

## Занятие 2. Алгоритмы с условием (Ветвления). Вычислить значение функции $y=f(x)$

### Задача

Дана следующая функция  $y=f(x)$ :

$$y = 2x - 10, \text{ если } x > 0$$

$$y = 0, \text{ если } x = 0$$

$$y = 2 * |x| - 1, \text{ если } x < 0$$

Требуется найти значение функции по переданному  $x$ .

### Пояснение к задаче и алгоритм решения

1. Получить с клавиатуры значение  $x$ .
2. Если  $x$  больше 0, то вычислить выражение  $2*x-10$ , результат присвоить переменной  $y$ .
  1. Иначе если  $x$  равен 0, то присвоить  $y$  значение 0.
  1. Иначе присвоить  $y$  результат выражения  $2*|x|-1$ .
3. Вывести значение  $y$  на экран.

### Исходный код на языке программирования Pascal

```
var x,y: integer;
begin
  readln(x);
  if x > 0 then y := 2*x-10
  else
  if x = 0 then y := 0
  else y := 2*abs(x)-1;
  writeln(y);
end.
```

## Занятие 3. Алгоритмы с условием (Ветвления). Проверка делимости одного числа на другое.

## Задача

Вводятся два целых числа. Проверить делится ли первое на второе. Вывести на экран сообщение об этом, а также остаток (если он есть) и частное (в любом случае).

## Пояснение к задаче и алгоритм решения

1. Если первое число нацело делится на второе, то вывести сообщение об этом.
2. Иначе вывести сообщение о том, что первое число не делится на второе, найти остаток от деления и также вывести его.
3. В конце программы найти частное от деления чисел и вывести его.

## Исходный код на языке программирования Pascal

```
var a,b: integer;  
begin  
  readln(a);  
  readln(b);  
  if a mod b = 0 then  
    writeln(a, ' делится на ', b)  
  else begin  
    writeln(a, ' НЕ делится на ', b);  
    writeln('Остаток: ', a mod b);  
  end;  
  writeln('Частное: ', a div b);  
end.
```

## Пример(ы) выполнения программы на языке Pascal

```
678  
4  
678 НЕ делится на 4  
Остаток: 2  
Частное: 169
```

```
680  
5  
680 делится на 5  
Частное: 136
```

**Занятие 4. Алгоритмы с условием (Ветвления). Перевести байты в килобайты или наоборот**

## Задача

Перевести число, введенное пользователем, в байты или килобайты в зависимости от его выбора.

### Пояснение к задаче и алгоритм решения

В данной задаче пользователь должен ввести два значения: само число и как-либо указать, во что он хочет его перевести (в байты или килобайты). Пусть выбор способа перевода указывается с помощью одного из двух символов. Например, если пользователь введет букву "b", то число будет переводиться в байты, а если букву "k", то в килобайты. Тогда алгоритм решения задачи будет следующим:

1. Ввести число.
2. Указать направление перевода.
3. Если выбран перевод в байты, то надо умножить число на 1024.
4. Если выбран перевод в килобайты, то надо разделить число на 1024.

В данной задаче нельзя оставлять второй случай без проверки, т.е. использовать исключительно ветку else, т.к. пользователь может ввести что угодно. Поэтому надо использовать либо две отдельные ветки if, либо вложенную в else вторую проверку через if. С точки зрения наиболее оптимального кода второй вариант предпочтительнее, т.к. в случае срабатывания первого if второй проверяться не будет.

Данную задачу проще решить с помощью оператор выбора case, а не if.

### Исходный код на языке программирования Pascal

```
var n: word;
c: char;
begin
  write('Число: ');
  readln(n);
  write('Перевести в байты(b) или килобайты(k): ');
  readln(c);
  if c = 'b' then
    writeln(n, 'Кб = ', n*1024, 'Б')
  else
    if c = 'k' then
      writeln(n, 'Б = ', n/1024:3:2, 'Кб');
end.
```

*// 2-й вариант (через оператор case):*

```
var
n: word;
c: char;
begin
write('Число: ');
readln(n);
write('Перевести в байты(b) или килобайты(k): ');
readln(c);
case c of
```

```
'b': writeln(n, 'Кб = ', n*1024, 'Б');  
'k': writeln(n, 'Б = ', n/1024:3:2, 'Кб');  
end;  
end.
```

### **Пример(ы) выполнения программы на языке Pascal**

Число: 5000

Перевести в байты(b) или килобайты(k): b  
5000Кб = 5120000Б

Число: 5000

Перевести в байты(b) или килобайты(k): k  
5000Б = 4.88Кб

## **Занятие 5. Алгоритмы с условием (Ветвления). Буква или иной символ?**

### **Задача**

Вводится целое число, обозначающее код символа по таблице ASCII. Определить, это

код английской буквы или какой-либо иной символ.

### Пояснение к задаче и алгоритм решения

В таблице ASCII символы букв идут по порядку - от "A" до "Z", далее через несколько небуквенных символов от "a" до "z".

Задача сводится к тому, что надо сравнить код-число, введенное пользователем, с кодами первых и последних букв алфавита. Если код принадлежит этим диапазонам, значит это код буквы, если нет - то небуквенный символ.

### Исходный код на языке программирования Pascal

```
var
  n, ba, bz, sa, sz: byte;
begin
  readln(n);
  ba := ord('A');
  bz := ord('Z');
  sa := ord('a');
  sz := ord('z');
  if (n >= ba) and (n <= bz) or (n >= sa) and (n <= sz) then
    writeln('Это буква ', chr(n))
  else
    writeln('Это не буква, а символ ', chr(n));
end.
```

### Пример(ы) выполнения программы на языке Pascal

64

Это не буква, а символ @

66

Это буква B

## Занятие 6. Алгоритмы с условием (Ветвления). Определить существование треугольника и его тип

### Задача

По длинам трех отрезков, введенных пользователем, определить возможность



существования треугольника, составленного из этих отрезков. Если такой треугольник существует, то определить, является ли он разносторонним, равнобедренным или равносторонним.

### Пояснение к задаче и алгоритм решения

Треугольник существует только тогда, когда сумма длин любых его двух сторон больше третьей стороны. Иначе две стороны просто "укладываются" на третьей.

Треугольник является разносторонним, если все его стороны имеют разную длину; треугольник будет равнобедренным, если любые две его стороны равны между собой, но отличны от третьей; и треугольник является равносторонним, когда все его стороны равны.

Прежде чем выяснять вид треугольника, необходимо удостовериться, что треугольник существует.

Если треугольник существует, то можно сначала проверить на неравенство три его стороны. Если они не равны друг другу, то треугольник **разносторонний**. Если это не так, то следующим шагом будет проверка на равенство всех сторон треугольника. Если все стороны равны, делается вывод о том, что треугольник **равносторонний**. Иначе остается только один вариант - **равнобедренный** треугольник.

### Исходный код на языке программирования Pascal

```
var
  a,b,c: word;
begin
  write('a='); readln(a);
  write('b='); readln(b);
  write('c='); readln(c);
  if (a+b <= c) or (a+c <= b) or (b+c <= a) then
    writeln('Такой треугольник не существует.')
  else
    if (a <> b) and (a <> c) and (b <> c) then
      writeln('Это разносторонний треугольник.')
    else
      if (a = b) and (b = c) then
        writeln('Это равносторонний треугольник.')
      else
        writeln('Это равнобедренный треугольник.');
```

### Пример(ы) выполнения программы на языке Pascal

a=34

b=6

c=5

Такой треугольник не существует.

a=7

b=8

c=9

Это разносторонний треугольник.

$a=6$

$b=6$

$c=4$

Это равнобедренный треугольник.

$a=5$

$b=5$

$c=5$

Это равносторонний треугольник.

## **Занятие 7. Алгоритмы с условием (Ветвления). Определение принадлежности точки кругу с центром в начале координат**

### **Задача**

Вводятся координаты  $(x; y)$  точки и радиус круга  $(r)$ . Определить принадлежит ли данная точка кругу, если его центр находится в начале координат.

## Пояснение к задаче и алгоритм решения

Будем считать, что точка принадлежит кругу, если находится внутри его или на его окружности.

Из любой точки координатной плоскости можно провести отрезок к началу координат. Если длина этого отрезка больше радиуса круга, то точка лежит за пределами круга и, следовательно, не принадлежит ему. Если же отрезок, соединяющий точку и начало координат, меньше радиуса круга с центром в начале координат или равен ему, то точка будет принадлежать кругу.

Отрезок между любой точкой и нулевой точкой (началом координат) является гипотенузой прямоугольного треугольника, катеты которого равны значениям  $x$  и  $y$  координаты данной точки.

Таким образом задача сводится по-сути к двум действиям:

1. Нахождение длины отрезка между точкой и началом координат по теореме Пифагора (квадрат длины гипотенузы равен сумме квадратов длин катетов).
2. Сравнению полученного значения с радиусом круга.

## Исходный код на языке программирования Pascal

```
var x,y,r,h: real;
begin
  write('координаты точки: ');
  readln(x,y);
  write('радиус круга: ');
  readln(r);
  h := sqrt(x*x + y*y);
  if h > r then writeln('Точка не принадлежит кругу')
  else writeln('Точка принадлежит кругу');
end.
```

## Пример(ы) выполнения программы на языке Pascal

```
координаты точки: -1 -2
радиус круга: 5
Точка принадлежит кругу
```

## Занятие 8. Алгоритмы с условием (Ветвления). Какой координатной четверти принадлежит точка?

### Задача

Определить четверть координатной плоскости, которой принадлежит точка. Координаты точки ввести с клавиатуры.

### Пояснение к задаче и алгоритм решения

- ⊙ Если у точки обе координаты ( $x$  и  $y$ ) положительны, то она принадлежит первой

четверти.

- ⓐ Если координата  $x$  отрицательна, а  $y$  положительна, то точка находится во второй четверти.
- ⓑ Если обе координаты отрицательны, то точка принадлежит третьей координатной четверти.
- ⓒ Если  $x$  положительна, а  $y$  отрицательна, то точка находится в IV четверти.

Следует иметь в виду, что использовать в программе четыре отдельные инструкции `if` не совсем правильно. Хотя такое решение даст верный результат, программу нельзя будет назвать эффективной, т.к. даже если первая проверка дала "правду", дальнейшие проверки будут продолжены, хотя в них нет никакого смысла. Поэтому правильным решением будет использование вложенных конструкций `if-else`. Это замечание не касается языка Python, т.к. в нем есть конструкция множественного ветвления (`if-elif-else`).

Поскольку точка может лежать на одной из двух координатных осей или находиться в начале координат, то значит могут быть ситуации, когда точка не принадлежит ни одной из четвертей. Эти случаи обрабатываются в отдельных ветках, либо опускаются. Из этого также следует, что если первые три проверки не сработали, то нельзя делать однозначный вывод, что точка принадлежит оставшейся четверти. Поэтому в программе сообщение о том, в какой четверти находится точка может быть только в теле `if`, но не `else`.

### Исходный код на языке программирования Pascal

```
var x,y: integer;
begin
  readln(x,y);
  if (x>0) and (y>0) then
    writeln('I quadrant')
  else
    if (x<0) and (y>0) then
      writeln('II quadrant')
    else
      if (x<0) and (y<0) then
        writeln('III quadrant')
      else
        if (x>0) and (y<0) then
          writeln('IV quadrant');
end.
```

### Пример(ы) выполнения программы на языке Pascal

```
-5 3
II quadrant
```

## Занятие 9. Алгоритмы с условием (Ветвления). Решить квадратное уравнение

### Задача

Найти корни квадратного уравнения и вывести их на экран, если они есть. Если корней нет, то вывести сообщение об этом. Конкретное квадратное уравнение определяется коэффициентами  $a$ ,  $b$ ,  $c$ , которые вводит пользователь.

### Пояснение к задаче и алгоритм решения

Квадратное уравнение имеет вид  $ax^2 + bx + c = 0$ . Коэффициенты  $a$ ,  $b$  и  $c$  - это

конкретные числа, а  $x$  надо найти, решив уравнение.

1. Вычислить дискриминант по формуле  $d = b^2 - 4ac$ .
2. Если дискриминант больше нуля, то вычислить два корня уравнения:  
 $x_1 = (-b + \sqrt{d}) / 2a$   
 $x_2 = (-b - \sqrt{d}) / 2a$
3. Если дискриминант равен нулю, то вычислить только один корень (второй будет равен ему).
4. Если дискриминант отрицателен, то вывести сообщение, что корней нет.

### Исходный код на языке программирования Pascal

```
var
a,b,c,d,x1,x2: real;
begin
write('a='); readln(a);
write('b='); readln(b);
write('c='); readln(c);
d := b*b - 4*a*c;
if d > 0 then begin
x1 := (-b + sqrt(d)) / (2*a);
x2 := (-b - sqrt(d)) / (2*a);
writeln('x1=',x1:3:2,'; x2=',x2:3:2);
end
else
if d = 0 then begin
x1 := (-b) / (2*a);
writeln('x=',x1:5:2);
end
else
writeln('Корней нет');
end.
```

### Пример(ы) выполнения программы на языке Pascal

```
a=7
b=1
c=-6
x1=0.86; x2=-1.00
```

### Занятие 10. Алгоритмы с условием (Ветвления). Определить високосный год или нет

#### Задача

Определить, является ли год, который ввел пользователем, високосным или невисокосным.

#### Пояснение к задаче и алгоритм решения

Високосные года делятся нацело на 4. Однако из этого правила есть исключение: столетия, которые не делятся нацело на 400, високосными не являются.

В високосном году 366 дней, в обычном 365.

1. Если год не делится на 4, значит он обычный.
2. Иначе надо проверить не делится ли год на 100.
3. Если не делится, значит это не столетие и можно сделать вывод, что год високосный.
4. Если делится на 100, значит это столетие и его следует проверить его делимость на 400.
5. Если год делится на 400, то он високосный.
6. Иначе год обычный.

Проверки можно проводить последовательно, а можно группировать через логические операторы "И" и "ИЛИ". Поэтому способов решения задачи может быть несколько.

### Исходный код на языке программирования Pascal

*// 1-й вариант:*

```
var y: word;
begin
  readln(y);
  if y mod 4 <> 0 then writeln('Обычный')
  else
    if y mod 100 = 0 then
      if y mod 400 = 0 then writeln('Високосный')
      else writeln('Обычный')
    else writeln('Високосный');
end.
```

*// 2-й вариант:*

```
var y: word;
begin
  readln(y);
  if (y mod 4 <> 0) or (y mod 100 = 0) and (y mod 400 <> 0) then
    writeln('Обычный')
  else
    writeln('Високосный');
end.
```

### Пример(ы) выполнения программы на языке Pascal

1980

Високосный    1978 Обычный

**Занятие 11. Алгоритмы с условием (Ветвления). Среди трех чисел найти среднее**

### Задача

Вводятся три разных числа. Найти, какое из них является средним (больше одного, но меньше другого).

### Пояснение к задаче и алгоритм решения

Проверить, лежит ли первое число между двумя другими. При этом может быть два случая:

- ⓐ первое больше второго и первое меньше третьего,
- ⓑ первое меньше второго и первое больше третьего.

Если ни один из вариантов не вернул истину, значит первое число не среднее. Тогда проверяется, не лежит ли второе число между двумя другими. Это может быть в двух случаях, когда

- ⓐ второе больше первого и меньше третьего,
- ⓑ второе меньше первого и больше третьего.

Если эти варианты также не вернули истину, то остается только один вариант - посередине лежит третье число.

### Исходный код на языке программирования Pascal

```
var
  a1, a2, a3: integer;
begin
  write ('Введите три числа: ');
  readln (a1, a2, a3);
  if ((a1 > a2) and (a1 < a3)) or ((a1 < a2) and (a1 > a3)) then
    writeln(a1)
  else
    if ((a2 > a1) and (a2 < a3)) or ((a2 < a1) and (a2 > a3)) then
      writeln(a2)
    else
      writeln(a3);
end.
```

### Пример(ы) выполнения программы на языке Pascal

Введите три числа: 3 5 4  
4

## Занятие 12. Алгоритмы с условием (Ветвления). Определить нечетное число

### Задача

Из двух случайных чисел, одно из которых четное, а другое нечетное, определить и вывести на экран нечетное число.

### Пояснение к задаче и алгоритм решения

В данной задаче можно выделить две подзадачи: 1) сгенерировать два случайных числа так, чтобы одно было четным, а другое нечетным, 2) определить, какое из них нечетное.

Вариант решения первой подзадачи:

Генерируем два случайных числа. Далее проверяем, являются ли оба числа четными или оба нечетными. Если это так, то увеличиваем первое число на 1. При этом в любом

случае одно станет четным, а другое - нечетным. Проверку осуществляем в заголовке оператора if, строя сложное логическое выражение.

Вариант решения второй подзадачи:

Также используем оператор ветвления. Если первое число нечетное, то выводим его, иначе выводим второе.

Проверка чисел на четность выполняется путем определения остатка от деления числа на 2. Если остаток равен нулю, значит, число четное. Если нет, то нечетное.

### Исходный код на языке программирования Pascal

```
var a, b: byte;
begin
  randomize;
  a := random(100);
  b := random(100);
  if ((a mod 2 = 0) and (b mod 2 = 0)) or ((a mod 2 <> 0) and (b mod 2 <> 0)) then
    a := a + 1;
  writeln(a, ', ', b);
  if a mod 2 <> 0 then
    writeln(a)
  else
    writeln(b);
end.
```

### Пример(ы) выполнения программы на языке Pascal

```
78 69
69
```

```
77 84
77
```