

Лабораторная работа 3

Тема: Программирование алгоритмов разветвляющейся структуры

Цель работы: научиться использовать при решении задач структуру ветвление.

Теоретическое введение и упражнения

Ветвление (развилка) – это алгоритмическая структура, в которой проверяется некоторое условие и в зависимости от результатов проверки выполняется то или иное действие.

Для программирования проверки условия и выбора действия в зависимости от этого условия используются **условные операторы**.

Условие в языке программирования может принимать два значения: истина (**True**) и ложь (**False**).

Условие записывается с помощью операций отношений и логических операторов, таблица 3.1.

Таблица 3.1 - Операции отношения

Оператор	Действие	Оператор	Действие
>	больше	<>	не равно
<	меньше	>=	больше или равно
=	равно	<=	меньше или равно

Использование операторов сравнения позволяет записывать простые условия. Из простых условий, которые являются выражениями логического типа, можно строить сложные условия с применением к ним, как к операндам, логических операторов.

Таблица 3.2 - Логические операции

Оператор	Действие	Оператор	Действие
and	логическое И	or	логическое ИЛИ
not	логическое НЕ	xor	логическое исключающее ИЛИ

На рисунке 3.1 приведены примеры задания условий с помощью операций отношений и логических операторов

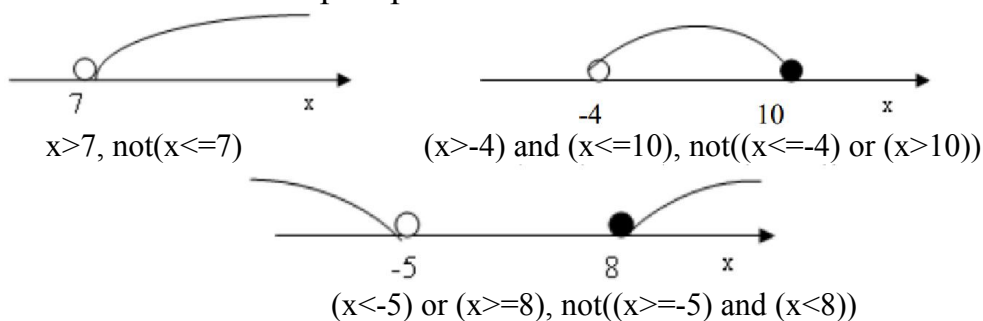


Рисунок 3.1 – Примеры условий

Условный оператор изменяет естественный порядок выполнения операторов программы. На рисунке 3.2 показан алгоритм работы условного оператора.

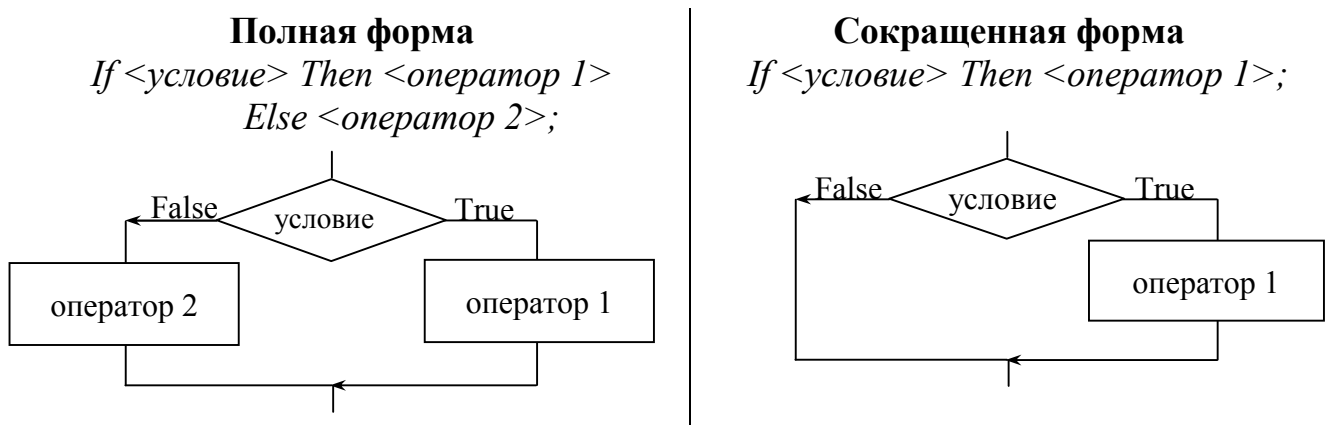


Рисунок 3.2 – Условный оператор If

Условие - это выражение или переменная булевского типа. Если условие имеет значение TRUE (т.е. истинно), тогда выполняется оператор 1, если FALSE (т.е. ложно) - оператор 2.

Точка с запятой после условия и оператора1, т.е. перед служебным словом ELSE, не ставится.

Пример.

```
If x <> 0
Then y := k/x
Else Label1.Caption := 'Деление на нуль запрещено';
```

Во втором случае если условие имеет значение TRUE (т.е. истинно), тогда выполняется оператор 1, если FALSE (т.е. ложно) - тогда выполняется оператор, следующий за оператором IF.

Пример.

```
If x > 0
Then y := sqrt(x);
```

Иногда после служебных слов Then или Else может стоять более одного оператора. Тогда необходимо использовать **составной оператор**.

Составной оператор может состоять из произвольного количества операторов, ограниченных словами Begin и End, разделенных между собой «;» (не путать с Begin и End, ограничивающими программу).

```
Begin
    Оператор1;
    Оператор2;
    .....
    Оператор n;
End;
```

Составной оператор воспринимается как *единое целое* и обычно используется в качестве *составной части других операторов*, где требуется применение нескольких операторов *вместо одного*. Для облегчения чтения программы рекомендуется располагать *Begin* и *End* на одной позиции в программе.

Пример.

If $x \neq 0$

Then

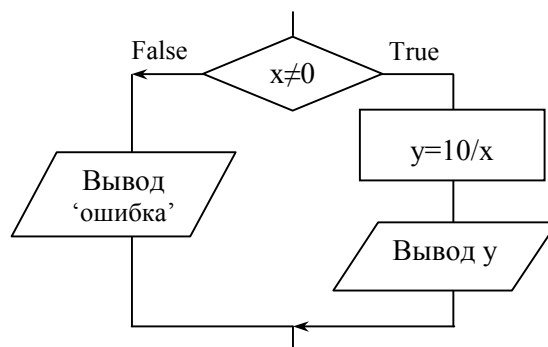
Begin

$y := 10/x;$

$Label1.Caption := FloatToStr(y);$

End

Else $Label1.Caption := 'Ошибка';$



Упражнение 1. Составить программу нахождения действительных и комплексных корней квадратного уравнения $ax^2 + bx + c = 0$.

Определим используемые в задаче переменные и их тип:

- входные переменные: a, b, c (коэффициенты уравнения);
 - промежуточные переменные: d (дискриминант);
 - результат: $x, x1, x2$ (корни уравнения).
- $a, b, c, d, x, x1, x2: real;$

Построим алгоритм решения задачи.

- 1) Ввод коэффициентов квадратного уравнения a, b и c .
- 2) Вычисление дискриминанта d по формуле $d = b^2 - 4ac$
- 3) Проверка знака дискриминанта.

Если $d \geq 0$, то корни уравнения действительные и находятся по формулам:

$$x_{1,2} = \frac{-b \pm \sqrt{d}}{2a}.$$

Если $d < 0$, то выводится сообщение «Корней нет».

Если $d = 0$, то корень уравнения действительный и находится по формуле:

$$x = \frac{-b}{2a}.$$

На основе вышеприведенного алгоритма постройте блок-схему.

На форме расположите компоненты как показано на рисунке 3.3.

Рисунок 3.3 – Окно формы для упражнения 1

Задайте свойства компонент в соответствии с таблицей 3.3.

Таблица 3.3 – Свойства компонент для упражнения 1

Название компонента	Свойство	Значение свойства
Form1	Caption	Решение квадратного уравнения
Label1	Caption	Введите коэффициенты уравнения
Label2	Caption	Коэффициент a=
Label3	Caption	Коэффициент b=
Label4	Caption	Коэффициент c=
Label5	Caption	Корни уравнения
Label6	Caption	x1=
Label7	Caption	x2=
Button1	Caption	Вычислить
Button2	Caption	Очистить
Edit1-5	Text	пусто

Процедура для вычисления значения функции будет иметь вид:

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
    var a,b,c,d,x,x1,x2:real;
```

```
    begin
```

```
        a:=strtofloat(edit1.text);
```

```
        b:=strtofloat(edit2.text);
```

```
        c:=strtofloat(edit3.text);
```

```
        d:=sqr(b)-4*a*c;
```

```
        if d>0
```

```
            then
```

```
                begin
```

```
                    x1:=(-b+sqr(d))/(2*a);
```

```
                    x2:=(-b-sqr(d))/(2*a);
```

```
                    edit4.text:=floattostr(x1);
```

```
                    edit5.text:=floattostr(x2);
```

```
                end
```

```
            else
```

```
                if d=0
```

```
                    then
```

```
                        begin
```

```
                            x:=-b/(2*a);
```

```
                            edit4.text:=floattostr(x);
```

```
                            label7.visible:=false;
```

```
                            edit5.visible:=false;
```

```
                        end
```

```
                    else
```

```
                        begin
```

```
                            label7.visible:=false;
```

```
                            edit4.text:='Решений нет';
```

```
                            label7.visible:=false;
```

```
                            edit5.visible:=false;
```

```
                        end;
```

```
    end;
```

Процедуру для кнопки «Очистить» написать самостоятельно.

Упражнение 2. Дано вещественное число x . Для функции, представленной графиком на рисунке 3.4, вычислить $y=f(x)$.

Определим переменные задачи и их тип:

входные переменные: x : *real*;

результат: y : *real*.

Аналитически функцию, представленную на рисунке 3.4 можно записать следующим образом:

$$y = \begin{cases} 4, & x \leq -2 \\ x^2, & -2 < x < 1 \\ 1, & x \geq 1 \end{cases}$$

На форме необходимо расположить компоненты и задать им соответствующие свойства, в соответствии с рисунком 3.5.

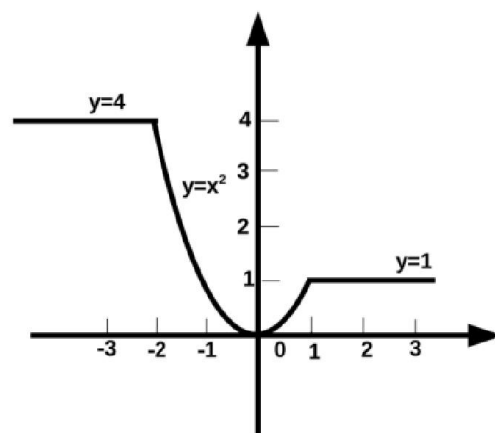


Рисунок 3.4 – График функции для упражнения 2

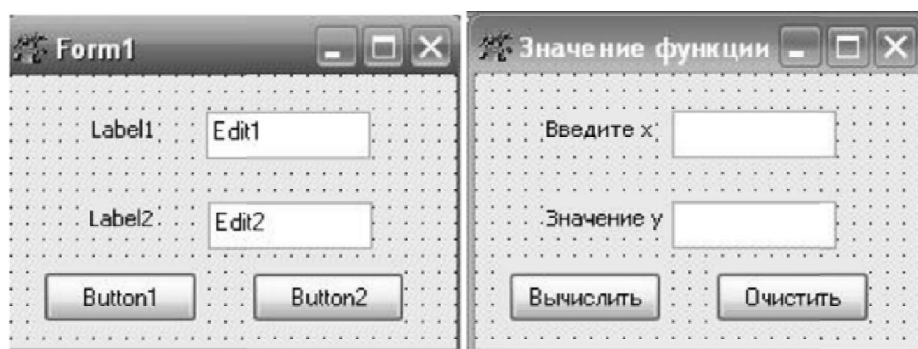


Рисунок 3.5 – Компоненты для упражнения 2

На рисунке 3.6 показан алгоритм решения задачи. Процедура для вычисления значения функции.

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var x,y:real;
```

```
begin
```

```
  x:=strtoint(edit1.text);
```

```
  if x<=-2
```

```
  then
```

```
    y:=4
```

```
  else
```

```
    if (x>1)
```

```
      then y:=1
```

```
    else y:=sqr(x);
```

```
    edit2.text:=floattostr(y);
```

```
end;
```

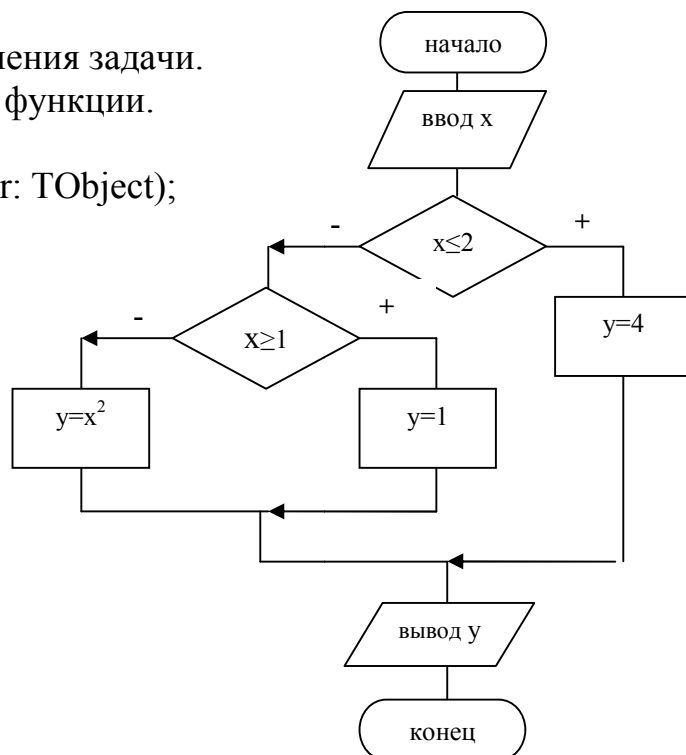



Рисунок 3.6 – Алгоритм для упражнения 2

Процедуру для кнопки «Очистить» написать самостоятельно.

Компонент CheckBox. При реализации ветвления кроме известных компонент используется компонента *CheckBox* (флажок, выключатель) . *CheckBox* позволяет пользователю выбрать/отменить определенную опцию.

Для задания состояния флажка используется свойство *State*.

CheckBox может находиться в общем случае в трех состояниях:

- флажок выбран, в окошке компонента стоит галочка. Этому состоянию соответствует значение *State=cbChecked*;
- флажок не выбран, в окошке компонента галочка отсутствует. Этому состоянию соответствует значение *State=cbUnchecked*;
- флажок выбран, в окошке компонента стоит галочка. Но само окошко и галочка серого цвета. Этому состоянию соответствует значение *State=cbGrayed*. Это, так называемое, промежуточное состояние. Свойство *AllowGrayed=[true, false]* включает (выключает) состояние *cbGrayed*.

Упражнение 3. Составить программу «Флажок» для вычисления квадрата и модуля числа.

Определим переменные задачи и их тип:

входные данные: *x: real*;

результат: *y1, y2: real* – модуль и квадрат числа соответственно.

Для решения задачи расположите компоненты на форме как показано на рисунке 3.7.



Рисунок 3.7 – Компоненты для упражнения 3

Компонентам задайте свойства как показано в соответствии с таблицей 3.4.

Таблица 3.4 – Свойства компонент для упражнения 3

Название компонента	Свойство	Значение свойства
Label1	Caption	Введите x
CheckBox1	Caption	Модуль
CheckBox2	Caption	Квадрат
Edit1	Text	пусто
Edit2	Text	пусто
Edit3	Text	пусто

В программе будет происходить два события: щелчок мыши по компоненте «Модуль» и щелчок мыши по компоненте «Квадрат». При этом будет обрабатываться событие OnChange для компонентов CheckBox1 и CheckBox2. Ниже показаны алгоритмы работы соответствующих процедур и сами процедуры.

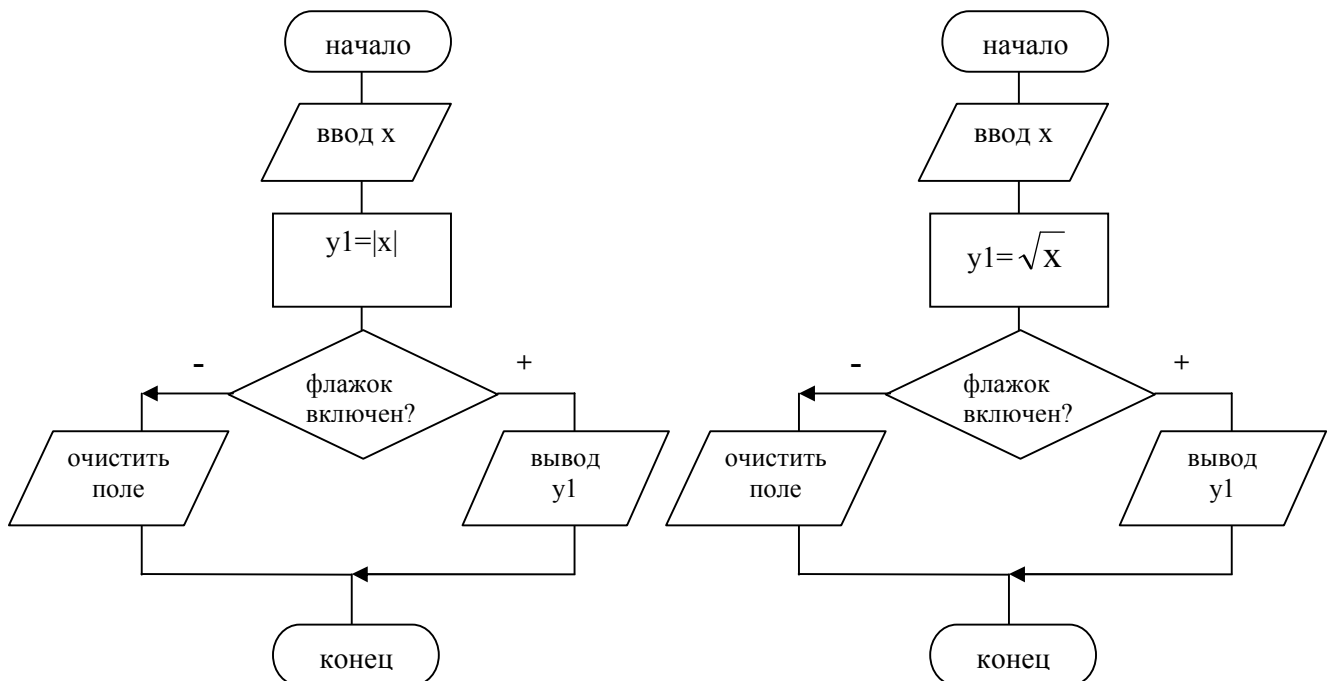


Рисунок 3.8 – Алгоритмы процедур для упражнения 3

```

procedure TForm1.CheckBox1Change();
    var x,y1:real;
begin
    x:=strtofloat(edit1.text);
    y1:=abs(x);
    if checkbox1.state=cbchecked
    then
        it2.text:=floattostr(y1)
    else edit2.clear;
end;

```

```

procedure TForm1.CheckBox2Change();
    var x,y2:real;
begin
    x:=strtofloat(edit1.text);
    y2:=sqr(x);
    if checkbox2.state=cbchecked
    then
        it3.text:=floattostr(y2)
    else edit3.clear;
end;

```

Наберите тексты процедур и отладьте программу.

Задания для самостоятельного выполнения

Задания выполняются по варианту. Вариант определяется по номеру компьютера.

Задание 1. Составить блок-схему и программу. Дано вещественное число x . Для функции, представленной графиком в таблице 3. 5, вычислить $y=f(x)$.

Таблица 3.5 – Варианты для задания 1

Вариант	Функция	Вариант	Функция
1		2	
3		4	
5		6	
7		8	
9		10	
11		12	

Задание 2. Составить блок-схему и программу для задачи соответствующей вашему варианту, таблица 3.6.

Таблица 3.6 - Варианты для задания 2

Вариант	Текст задачи			
1	Определить стоимость набора конфет в зависимости от введенного названия и веса.			
	Наименование	Цена за 1 кг,	Наименование	Цена за 1 кг, руб.
	Красная шапочка	350	Воронежские	240
	Алые паруса	380	Чародейка	270
2	Постройте модель анализа пожарного датчика в помещении, которая выводит сообщение 'Пожароопасная ситуация', если температура (в вашей модели она будет вводиться с клавиатуры) в комнате превысила 60°C.			
3	Из трех вводимых вами целых чисел выводится заключение о том, какое число больше.			
4	Рис расфасован в два пакета. Вес первого — m кг, второго — n кг. Вывести сообщение какой пакет тяжелее — первый или второй.			
5	Проверить, верно ли утверждение, что введенное вами целое число является четным.			
6	Проверить, верно ли утверждение, что введенное вами целое число делится без остатка на 3.			
7	Постройте модель для анализа возраста человека и отнесение его к одной из четырех групп: дошкольник, ученик, работник, пенсионер. Возраст вводится с клавиатуры.			
8	Определить, входит ли введенная вами цифра в десятичную запись введенного трехзначного числа.			
9	К финалу конкурса лучшего по профессии "Специалист электронного офиса" были допущены трое: Иванов, Петров, Сидоров. Соревнования проходили в три тура. Иванов в первом туре набрал m_1 баллов, во втором — n_1 , в третьем — p_1 . Петров — соответственно m_2 , n_2 , p_2 ; Сидоров — m_3 , n_3 , p_3 . Определить, сколько баллов набрал победитель.			
10	Постройте модель реализующую эпизод применения компьютера в книжном магазине. Компьютер запрашивает стоимость книг, сумму денег, внесенную покупателем; если сдачи не требуется, печатает на экране "спасибо"; если денег внесено больше, то печатает "возьмите сдачу" и указывает сумму сдачи; если денег недостаточно, то печатает это сообщение и указывает размер недостающей суммы			
11	Постройте модель, которая спрашивает возраст человека и, если ему 18 лет и больше, сообщает "Замечательно. Вы уже можете водить автомобиль", а в противном случае – "К сожалению, водить автомобиль Вам рановато".			
12	Постройте модель, определяющую делится ли нацело введенное вами трехзначное число на сумму своих цифр.			

Контрольные вопросы

- 1 Что представляет собой ветвление как алгоритмическая структура?
- 2 Перечислите известные Вам операции отношения и логические операции. Приведите примеры их использования.
- 3 Перечислите действия, реализуемые при выполнении условного оператора?

- 4 Какие виды оператора IF существуют?
- 5 Когда применяется сокращенная форма оператора IF?
- 6 Когда применяется полная форма оператора IF?
- 7 Как организовать разветвление вычислений: а) на две ветви; б) на три ветви?
- 8 Для чего используется составной оператор? Приведите пример.
- 9 Как в программе можно использовать компонент CheckBox?