

ЛАБОРАТОРНАЯ РАБОТА №3. ПРОГРАММИРОВАНИЕ АЛГОРИТМОВ РАЗВЕТВЛЯЮЩЕЙСЯ СТРУКТУРЫ

1. ЦЕЛЬ РАБОТЫ

Целью лабораторной работы является приобретение умений и навыков практического освоения современных технологий визуального объектно-ориентированного программирования в свободно распространяемой среде разработки приложений Lazarus на примере создания приложений разветвляющейся структуры с использованием условных операторов и операторов выбора языка программирования FREE Pascal.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1. СОСТАВНОЙ ОПЕРАТОР

Составной оператор представляет собой группу из произвольного числа операторов, ограниченную операторными скобками :

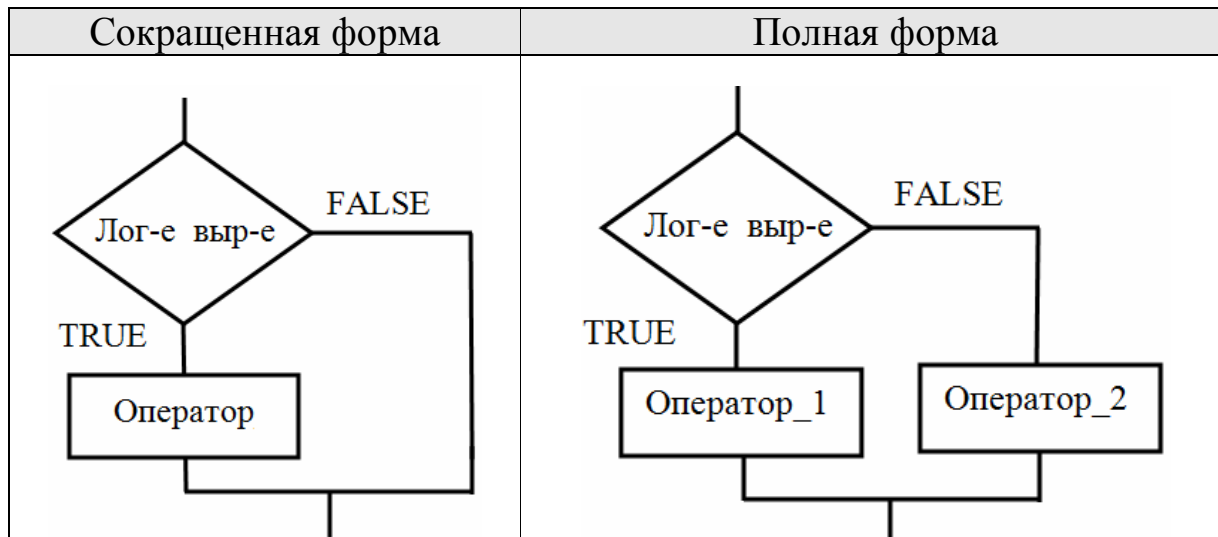
```
Begin  
  <оператор 1>;  
  <оператор 2>;  
  -----  
  <оператор N>;  
End;
```

Транслятор воспринимает составной оператор как один оператор.

2.2. УСЛОВНЫЙ ОПЕРАТОР

Выбор одного из альтернативных путей работы программного кода в зависимости от результата проверки какого-либо условия (логического выражения) обеспечивают условные операторы.

Условный оператор существует в двух формах: полной и сокращенной, и схематично может быть изображен следующим образом:



Формат условного оператора:

Сокращенная форма:

```
If <логическое выражение> Then <Оператор>;
```

Полная форма:

```
If <логическое выражение> Then <Оператор_1>  

Else <Оператор_2>;
```

Замечание. Перед Else точка с запятой не ставится!

Каждый из операторов, входящих в состав условного оператора может быть простым или составным.

Кроме того, один условный оператор может входить в состав другого, например:

```
If <лог_выр_1> Then  

    If <лог_выр_2> Then <Оператор_1>  

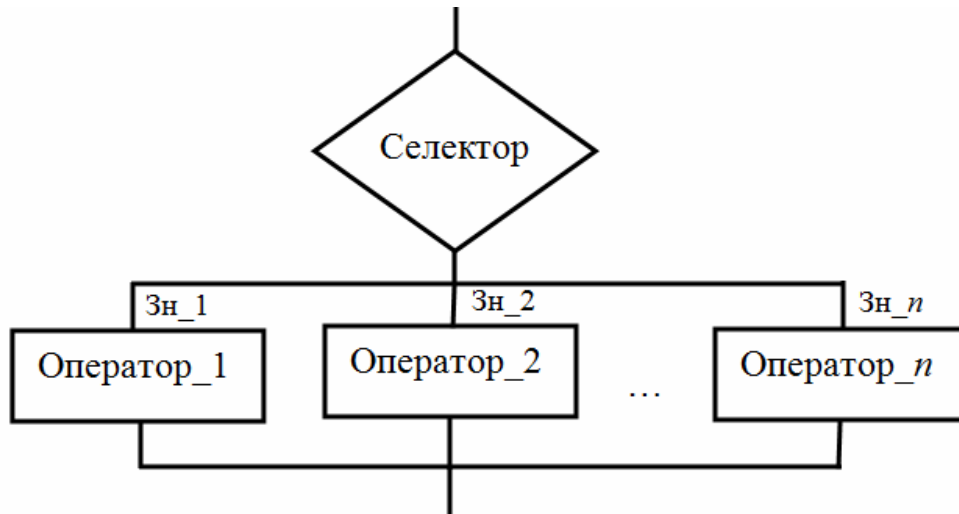
    Else <Оператор_2>  

Else <Оператор_3>;
```

Замечание. При вложенности условных операторов каждое Else соответствует тому Then, которое непосредственно ему предшествует и еще не закрыто.

2.3. ОПЕРАТОР ВЫБОРА CASE

Оператор `Case` дает возможность выполнить один из нескольких возможных операторов в зависимости от некоторого выражения, называемого *селектором*. Оператор `Case` выбирает для исполнения тот оператор, одна из констант выбора которого равна текущему значению селектора:



Формат оператора выбора:

```
Case <k> of  
  <A1>: <Оператор_1>;  
  <A2>: <Оператор_2>;  
  . . . . .  
  <AN>: <Оператор_N>;  
  [Else <Оператор_N1>]  
End;
```

Селектор `k` может иметь только простой порядковый тип (целый, символьный, логический). Константы `A1`, ..., `AN` должны быть того же типа, что и селектор.

Исполнение оператора `Case` начинается с вычисления значения селектора. Затем обеспечивается реализация того оператора, константа выбора которого равна текущему значению селектора. Если ни одна из констант не равна текущему значению селектора, выполняется оператор, стоящий за словом `Else`. Если слово `Else` отсутствует, активизируется оператор, следующий за оператором `Case`.

Замечание 1. Если при выборе альтернативы необходимо выполнение нескольких операторов используйте составной оператор.

Пример:

```
Ost:=c mod x;  
Case ost of  
  0:      Begin c:=c div x; Text1.Text:="Остаток 0"; end;  
  2,4..8: c:=c-ost;  
  1,3:    c:=c+ost;  
end;
```

Замечание 2. Не забывайте закрывать оператор Case служебным словом end!!!

3. ПРАКТИЧЕСКАЯ ЧАСТЬ

3.1. ПРИМЕРЫ ПРИЛОЖЕНИЙ РАЗВЕТВЛЯЮЩЕЙ СТРУКТУРЫ

Используя компоненты класса TEdit и TButton, самостоятельно создайте формы для примеров, приведенных ниже. Постарайтесь написать событийные процедуры самостоятельно. Сравните свой текст с приведенным в практикуме. Не забудьте сохранять *каждый* проект в *новую* папку. Запустите созданные приложения на выполнение. Проанализируйте результат.

Пример 1. Найти максимальное из трех действительных чисел.

Код событийной процедуры:

```
Var  
  Max, a,b,c: real;  
Begin  
  a:=StrToFloat(Edit1.text); {ввод трех действительных чисел}  
  b:=StrToFloat(Edit2.text);  
  c:=StrToFloat(Edit3.text);  
  if a >=c then Max:=a else Max:=c  
    else if b >= c then Max:=b else Max:=c;  
  Edit3.text:=FloatToStr(Max);  
End;
```

Пример 2. Вычислить значение кусочно-заданной функции для заданного значения аргумента:

$$y = \begin{cases} t^{2a \ln t}, & \text{для } 1 \leq t \leq 2 \\ 1, & \text{для } t < 1 \\ e^t \cos at, & \text{для } t > 2 \end{cases}$$

Код событийной процедуры:

```
Procedure TForm1.Button1Click(Sender: TObject);
Var
  y,t: real;
Begin
  t:=StrToFloat(Edit1.text); {ввод аргумента}
  a:=StrToFloat(Edit2.text); {ввод параметра a}
  If t <1 then y:=1
    else if t >2 then y:=exp(t)*cos(a*t)
      else y:=Power(t,2*a*ln(t));
  Edit3.text:=FloatToStr(y);
End;
```

Замечание. Не забудьте подключить модуль Math, в котором находится функция возведения степень Power.

В результате выполнения проекта получается результат с 13-ю знаками после запятой:

Вычисление значения кусочно-заданной функции

$$y = \begin{cases} t^{2a \ln t}, & \text{для } 1 \leq t \leq 2 \\ 1, & \text{для } t < 1 \\ e^t \cos at, & \text{для } t > 2 \end{cases}$$

Введите t

Введите a

Результат

Однако в Lazarus есть функция позволяющая выполнить форматный вывод:

`FloatToStrF(x, Format, Precision, Digits),`

здесь `x` – вещественное число,
`Precision` – общее количество цифр в мантиссе,
`Digits` – количество цифр в десятичном порядке.

Правила использования параметров `Format`, `Precision` и `Digits` функции `FloatToStrF` приведены в табл. 3.

Таблица 3

Значение параметра <code>Format</code>	Описание
<code>ffExponent</code>	Отображает число в экспоненциальной форме <code>Precision</code> –общее количество цифр в мантиссе; <code>Digits</code> –количество цифр в десятичном порядке
<code>ffFixed</code>	Отображает число в формате с фиксированной запятой. <code>Precision</code> –общее количество цифр; <code>Digits</code> –количество цифр в дробной части

Примеры строк программного кода с выводом чисел по формату:

Строка программного кода	Результат
<code>Edit1.Text:=FloatToStrF(s, ffExponent, 8, 3);</code>	1,0000780E+004
<code>Edit1.Text:=FloatToStrF(s, ffFixed, 8, 3);</code>	10000,780

Задание. Замените функцию `FloatToStr` на `FloatToStrF` в примере 2. Проанализируйте вид результата.

3.2. ВВОД ИНФОРМАЦИИ





Пример 3. Создайте проект решения квадратного уравнения $ax^2 + bx + c = 0$ с использованием условного оператора.


Код событийной процедуры:

```
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:=b*b - 4*a*c; {Вычисление определителя}  
  If d<0 Then Edit4.Text:='Действительных корней нет'  
    Else  
      if d = 0 Then  
        Begin  
          x:=-b/(2*a);  
          Edit4.Text:='Уравнение имеет один корень '+  
                    FloatToStrF(x, ffFixed,7,5);  
        End  
          Else  
            Begin  
              x1:= (-b + sqr(d)) / (2 * a);  
              x2:= (-b - sqr(d)) / (2 * a);  
              Edit4.Text:='Уравнение имеет два корня:  
              x1=' + FloatToStrF(x1, ffFixed,7,4) +  
              '; x2= ' + FloatToStrF(x2, ffFixed,7,4);  
            End;  
        End;  
  End;
```

3.3. КОМПОНЕНТЫ LAZARUS, ИСПОЛЬЗУЕМЫЕ ДЛЯ ПОСТРОЕНИЯ УПРАВЛЯЮЩИХ КОНСТРУКЦИЙ

Наиболее часто используемые компоненты группы Standard для построения разветвляющих конструкций приведены в табл. 4.

Значок	Класс	Название
	TRadioButton	Переключатель
	TCheckBox	Флажок
	TRadioGroup	Группа связанных переключателей
	TCheckBoxGroup	Группа связанных флажков

Компонент `RadioButton`  (Переключатель) используется для выбора одного из нескольких взаимоисключающих решений. Обычно на форму помещается, по меньшей мере, два таких компонента. Они могут иметь только два состояния, определяемых свойством `Checked`. Если у одного из компонентов это свойство истинно, то у всех остальных ложно.

Основные свойства `RadioButton`:

Свойство	Описание
<code>Caption</code>	Задаёт сопровождающий переключатель текст
<code>AllowGrayed</code>	Задаёт наличие у кнопки третьего состояния. Если значение этого свойства <code>False</code> , то кнопка может находиться в двух состояниях – включенном или выключенном. Если значение свойства равно <code>True</code> , то возможно третье состояние, когда кнопка неактивна.
<code>Checked</code>	Возвращает значение <code>True</code> , если кнопка включена, <code>False</code> – выключена.

Задание 1. Откройте проект примера 3 (решение квадратного уравнения). Добавьте на форму два переключателя с надписями `Форматный вывод` и `Бесформатный вывод` соответственно. Вставьте в исходный код событийной процедуры реализацию форматного или бесформатного вывода найденных корней в зависимости от включенного переключателя (свойство `Checked` соответствующего `RadioButton`).

Фрагмент кода, реализующего Форматный или Бесформатный вывод в случае одного корня:

```
Edit4.Text:='Уравнение имеет один корень ' ;  
if RadioButton1.Checked then  
    Edit4.Text:=Edit4.Text+FloatToStrF(x, ffFixed, 7, 5)  
if RadioButton2.Checked then  
    Edit4.Text:=Edit4.Text+FloatToStr(x);
```

Пример формы решения квадратного уравнения с реализацией форматного вывода:


Решение квадратного уравнения

a= 3.765475786 b= 7.8479837 c= 2.7856

Форматный вывод Бесформатный вывод Вычислить

Уравнение имеет два корня: x1=50,1480; x2= -52,2322

Закреть

Компонент CheckBox  (флажок) используется для того, чтобы пользователь мог включить или выключить значение какой-либо опции. Установлен флажок или нет определяет свойство `Checked`. В составе диалогового окна может быть несколько таких компонентов, причем состояние любого из них не зависит от состояния остальных. Относится к группе `Standard`.
Основные свойства компонента `CheckBox` :

Свойство	Описание
Caption	Задаёт сопровождающий флажок текст
AllowGrayed	Задаёт наличие у флажка третьего состояния. Если значение этого свойства False, то флажок может находиться в двух состояниях – включенном или выключенном. Если значение свойства равно True, то возможно третье состояние, когда флажок неактивен.
Checked	Возвращает или задаёт наличие галочки у флажка. (True, False)

Задание 2. Продолжаем усовершенствовать проект решения квадратного уравнения: будем вычислять комплексные корни по желанию пользователя. Добавьте на форму флажок с надписью Выводить комплексные корни. Вставьте в исходный код событийной процедуры реализацию вычисления комплексных корней в случае, если свойство Checked компоненты CheckBox имеет значение True.

Фрагмент кода, реализующего вычисление комплексных корней:

```
If CheckBox1.Checked Then
  Begin
    x1:=-b/(2*a);
    x2:=sqrt(abs(d))/(2*a);
    Edit4.Text:='x1=' + FloatToStrF(x1,ffFixed,7,4)+
      ' + i*' + FloatToStrF(x2,ffFixed,7,4);
    Edit4.Text:=Edit4.Text + ';    x2='+
      FloatToStrF(x1,ffFixed,7,4) +
      ' + i*' + FloatToStrF(x2,ffFixed,7,4);
  End;
```

Окончательная форма и код событийной процедуры представлены ниже (рис. 10, 11, 12):

```

Var
a, b, c, d, x, x1, x2: Real;
Begin
a:=StrToFloat (Edit1.Text);
b:=StrToFloat (Edit2.Text);
c:=StrToFloat (Edit3.Text);
d:=b*b - 4*a*c; {Вычисление определителя}

If d<0 Then

    If CheckBox1.Checked Then
        Begin
            x1:=-b/(2*a);
            x2:=sqrt(abs(d))/(2*a);
            Edit4.Text:='x1='+ FloatToStrF(x1, ffFixed, 7, 4)+
                ' + i*' + FloatToStrF(x2, ffFixed, 7, 4);
            Edit4.Text:=Edit4.Text +';    x2='+
                FloatToStrF(x1, ffFixed, 7, 4) +
                ' + i*' + FloatToStrF(x2, ffFixed, 7, 4);
        End
    else

        Edit4.Text:='Действительных корней нет'

    Else

        if d = 0 Then
            Begin
                x:=-b/(2*a);
                Edit4.Text:='Уравнение имеет один корень ';
                if RadioButton1.Checked then
                    Edit4.Text:=Edit4.Text+FloatToStrF(x, ffFixed, 7, 5);
                if RadioButton2.Checked then
                    Edit4.Text:=Edit4.Text+FloatToStr(x);
            End

            Else
                Begin
                    x1:= (-b + sqr(d)) / (2 * a);
                    x2:= (-b - sqr(d)) / (2 * a);
                    Edit4.Text:='Уравнение имеет два корня:  x1=';
                    if RadioButton1.Checked then Edit4.Text:=Edit4.Text +
                        FloatToStrF(x1, ffFixed, 7, 4) + ';    x2= ' +
                        FloatToStrF(x2, ffFixed, 7, 4);
                    if RadioButton2.Checked then
                        Edit4.Text:=Edit4.Text +
                            FloatToStr(x1) + ';    x2= ' + FloatToStr(x2);
                End;
            End;

```

ВЫЧИСЛЯЮТСЯ
КОМПЛЕКСНЫЕ
КОРНИ

Рис. 10. Код событийной процедуры

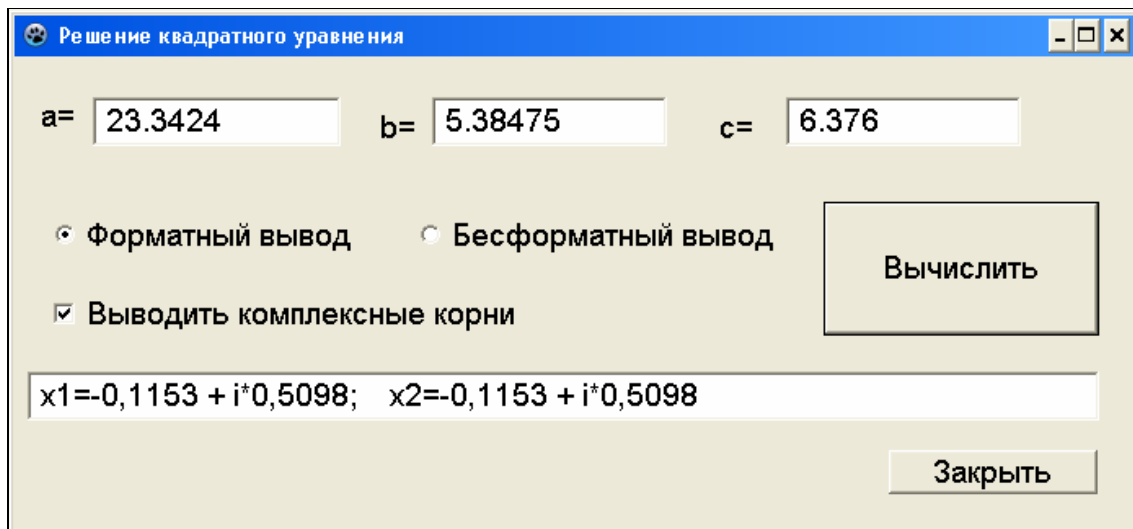


Рис. 11. Комплексные корни

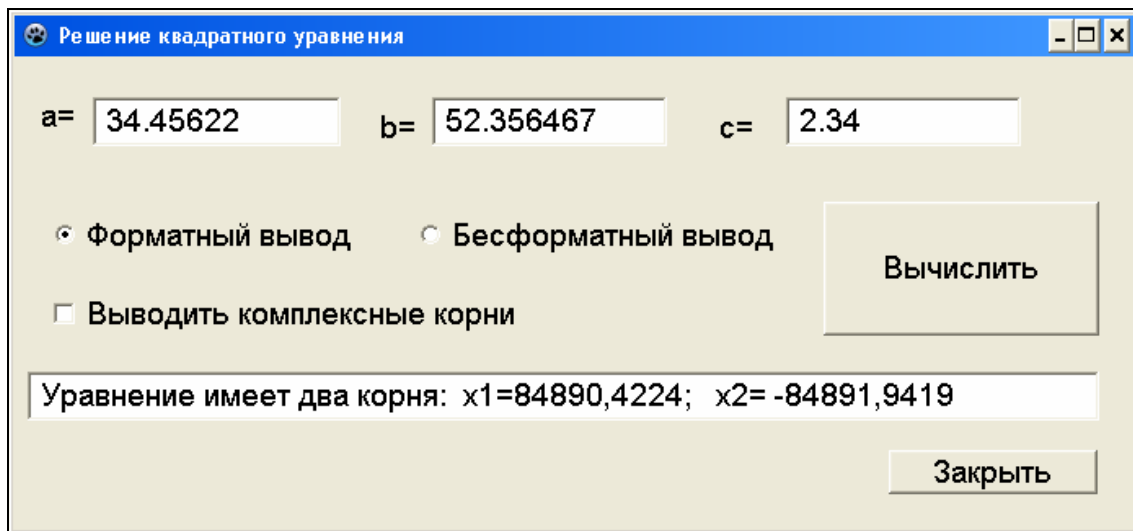



Рис. 12. Действительные корни



Компонент `RadioGroup` (Группа переключателей)  объединяет в себе несколько связанных переключателей.

Основные свойства компонента `RadioGroup`:

Свойство	Описание
<code>Caption</code>	Задаёт название группы переключателей
<code>Items</code>	Определяет количество переключателей в группе и надписи около них
<code>ItemIndex</code>	Задаёт номер кнопки, выбранной по умолчанию: 0 – первая, -1 – ни одна кнопка не выбрана

Пример 3. Создайте проект «Калькулятор»

Следуйте рекомендациям, описанным ниже.

1. Создайте новый проект. На форму поместите 3 компонента класса TEdit для ввода двух чисел и вывода результата и две кнопки для выхода из проекта.
2. Разместите на форме Компонент RadioGroup . После размещения на форме компонент пуст. Чтобы создать в нем хотя бы один переключатель, нужно выделить его и в Инспекторе объектов выбрать свойство Items. Щелкните на значок  (рис. 13). Откроется Диалог ввода строк – это редактор списка. Строки, набранные в редакторе, используются как поясняющие надписи справа от переключателей, а их количество определяет количество переключателей в группе (рис. 14).

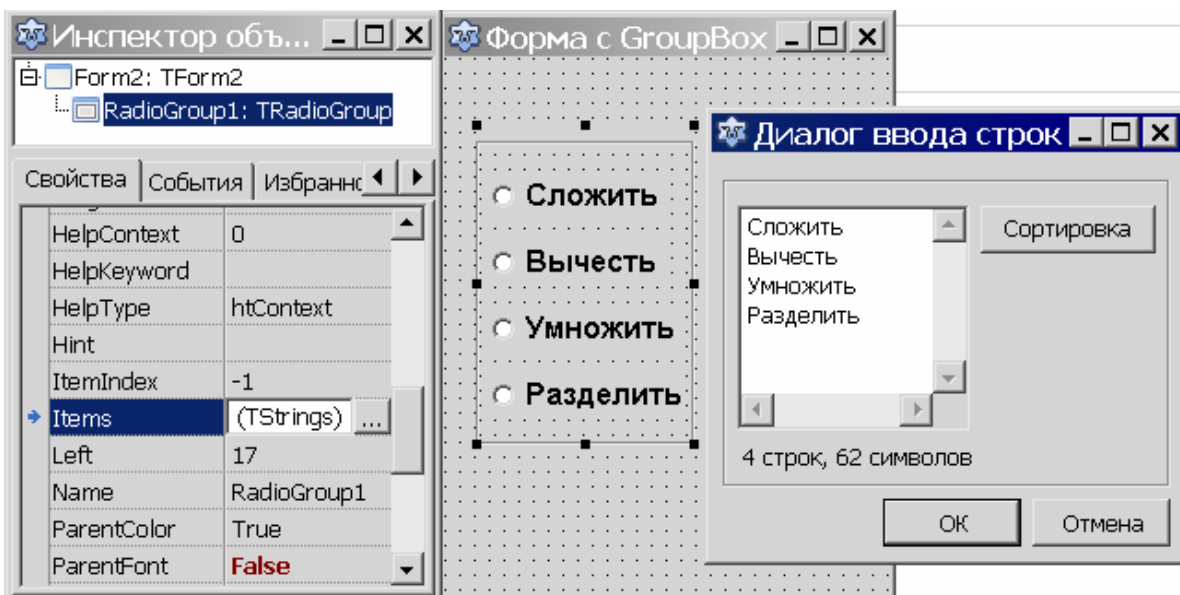


Рис. 13. Формирование списка переключателей на форме

3. После создания компонента RadioGroup его свойство ItemIndex по умолчанию равно -1. Это означает, что ни один компонент в группе не установлен. Чтобы в момент появления компонента на экране один из переключателей был отмечен, нужно либо на этапе конструирования формы, либо программно присвоить свойству ItemIndex номер одного из элементов списка, учитывая, что нумерация начинается с нуля. Присвойте свойству ItemIndex значение 0.

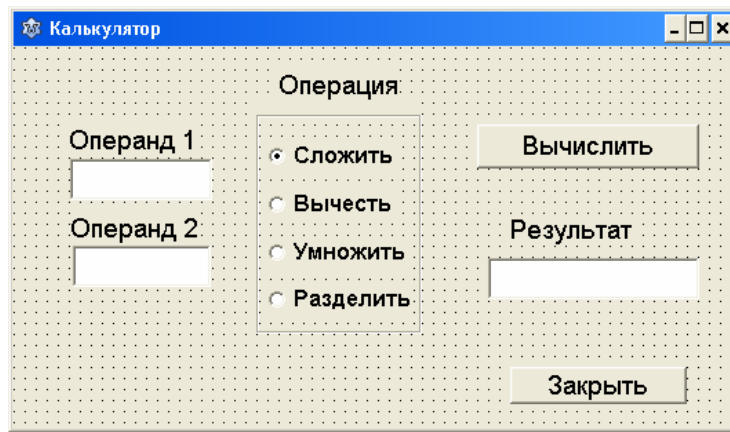


Рис. 14. Форма «Калькулятор»

4. Щелкните дважды на кнопке «Вычислить», чтобы открыть событийную процедуру. Создайте код процедуры. Используйте оператор Case для выбора варианта вычислений в зависимости от того, какой из переключателей компоненты RadioGroup отмечен. Запустите проект на выполнение.
5. Добавьте на форму флажок CheckBox с надписью Форматный/Бесформатный вывод (рис. 15). Вставьте в исходный код реализацию форматного вывода в случае, если свойство Checked компоненты CheckBox имеет значение True.

```

Procedure TForm1.Button1Click(Sender: TObject);
Var a,b,c:real;
Begin
    a:=StrToFloat(Edit1.Text);
    b:=StrToFloat(Edit2.Text);
    Case RadioGroup1.ItemIndex of
        0: c:=a+b;
        1: c:=a-b;
        2: c:=a*b;
        4: c:=a/b;
    End;
    If CheckBox1.Checked then
        Edit3.Text:=FloatToStr(c)
        else
        Edit3.Text:=FloatToStrF(c,ffFixed,7,2);
    End;

```

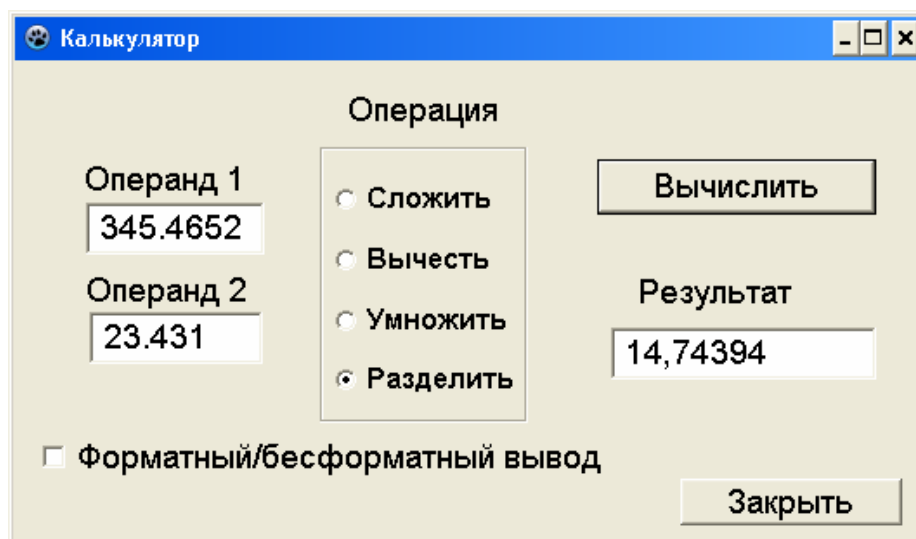


Рис. 15. Форма «Калькулятор» с компонентой CheckBox

6. Введите для Операнда 2 значение, равное нулю и выберите операцию Разделить. Посмотрите, какую ошибку выведет транслятор. В нашем проекте мы не учли, что пользователь может ввести значение равное нулю для делителя.

Создавая программу, необходимо не только реализовать алгоритм решения поставленной задачи, но и предугадать ошибки, которые может допустить пользователь, например, при вводе данных. Если пользователь допустил ошибку, необходимо проинформировать его об этом.

3.4. ОБРАБОТКА ОШИБОК. ВЫВОД СООБЩЕНИЙ В LAZARUS

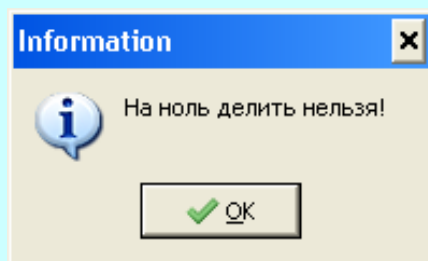
Для выдачи сообщения пользователю в Lazarus имеются следующие подпрограммы, которые выводят сообщения в отдельных окнах, не требующих проектирования:

Название	Описание
Процедура: ShowMessage ()	Выводит информационное сообщение в окне с одной кнопкой ОК.
Функция: MessageDlg ()	Выводит сообщения разных типов (информационное, сообщение об ошибке и др.) в окне с указанными типами кнопок. Возвращаемое функцией значение: нажатая кнопка.

Если диалоговое окно используется просто как некоторое сообщение пользователю, т.е. не нужно анализировать какая из его кнопок была нажата, то для его вызова можно использовать процедуру `ShowMessage (<Сообщение>)`.

```
ShowMessage ('На ноль делить нельзя! ');
```

Задание. Продолжаем усовершенствовать проект «Калькулятор». Откройте проект. Вставьте в исходный код вывод сообщения «На ноль делить нельзя», если параметр $b=0$. Запустите проект на выполнение. Выберите операцию деления и введите для делителя значение ноль. В результате вы увидите окно:



Часто возникают ситуации, когда необходимо выводить сообщения разных типов и при этом нужно узнать реакцию пользователя на это сообщение. В этом случае используется функция `MessageDlg()`, которая позволяет в окно сообщения помещать не только кнопку ОК, но и другие типы кнопок (ОК, ОТМЕНА, ДА, НЕТ и др.). При этом можно проанализировать какая из кнопок была нажата.

Формат вызова функции:

```
MessageDlg (Сообщение, Тип_сообщения,  
            [Список_кнопок], Справка),
```

где

Сообщение – строковое выражение, которое является сообщением или вопросом, обращенным к пользователю;

Тип сообщения – определяет внешний вид окна (см. табл. 5);

Список кнопок – константы, определяющие тип кнопок окна сообщения (см. табл. 6). Константы перечисляются через запятую в квадратных скобках;

Справка – номер окна справочной системы, которое будет выведено на экран, если нажать F1. Если использование справки не предусмотрено, то этот параметр равен нулю.

Таблица 5. Типы сообщения





Параметр	Тип окна	Значок в окне
mtInformation	Информационное (Information)	Значок информационного сообщения 
mtError	Сообщение об ошибке (Error)	Значок критического состояния 
mtConfirmation	Запрос на подтверждение (Confirmation)	Значок вопроса 
mtWarning	Предупредительное (Warning)	Значок предупреждающего сообщения 

Таблица 6. Типы кнопок

Наименование константы	Кнопка	Тип кнопки в диалоговом окне сообщений
mbYes	Да	
mbNo	Нет	
mbOk	Ok	
mbCancel	Отмена	
mbAbort	Стоп	
mbRetry	Повторить	
mbIgnore	Пропустить	

В зависимости от выбора пользователем кнопки функция MessageDlg возвращает следующие константы:

Нажатая кнопка	Константа
Нет	mrNo
Да	mrYes
Ok	mrOk
Отмена	mbCancel
Стоп	mbAbort
Повторить	mbRetry
Пропустить	mbIgnore

Задание. Откройте проект «Калькулятор». Создайте событийную процедуру для кнопки Закреть, в которой вызывается окно сообщений с двумя кнопками Yes и No и сообщением «Вы действительно хотите выйти из программы? Завершить работу сейчас?»

На рис. 16 представлен пример работы процедуры, вызывающей окно многострочного сообщения с двумя кнопками Yes и No:

```

Procedure TForm1.Button2Click(Sender: TObject);
Begin
  if MessageDlg('Выход из программы', 'Вы действительно
    хотите выйти из программы?' + Chr(13) + Chr(10) +
    'Завершить работу сейчас?', mtConfirmation,
    [mbYes, mbNo], 0) = mrYes
  then Close;
end;

```

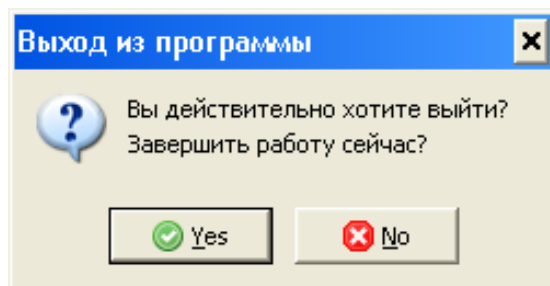


Рис. 16. Пример вызова функции MessageDlg()

4. ЗАДАНИЯ К ЛАБОРАТОРНОЙ РАБОТЕ №3

4.1. ПЕРВЫЙ УРОВЕНЬ СЛОЖНОСТИ

Задание 1. Вычислить значение кусочно-заданной функции для заданного аргумента

№ вар.	Функция	Диапазон изменения аргумента
1.	$y = \begin{cases} \sin x \lg x, & x > 3,5 \\ \frac{\sin x \cdot \cos x}{\lg x}, & 3,5 \geq x > 2 \\ \cos^2 x, & x \leq 2 \end{cases}$	[1; 5]
2.	$y = \begin{cases} \sin(x^2), & x \geq 1,5 \\ \lg(x+1), & 1,5 > x > 1 \\ \sin^2 \sqrt{ 20x }, & x \leq 1 \end{cases}$	[0,5; 2]
3.	$z = \begin{cases} \frac{1+ x }{\sqrt[3]{1+x^2+x}}, & x \leq -1, \\ 2 \ln(1+x^2) + \frac{1+\cos^4(x)}{2+x}, & x \in (-1,0), \\ (1+x)^{3/5}, & x \geq 0. \end{cases}$	[-2; 2]
4.	$z = \begin{cases} 3x + \sqrt{1+x^2}, & x < 0, \\ 2 \cos(x)e^{-2x}, & x \in [0, 1], \\ 2 \sin(3x) & x > 1. \end{cases}$	[-2; 1,5]
5.	$z = \begin{cases} \sqrt{1 + \frac{x^2}{1+x^2}}, & x < 0, \\ 2 \cos^2(x), & x \in [0, 1], \\ \sqrt{1 + 2 \sin(3x) ^{1/3}}, & x > 1. \end{cases}$	[-1,5; 1,5]

6.	$z = \begin{cases} x ^{1/3}, & x < 0, \\ -2x + \frac{x}{1+x}, & x \in [0,1), \\ \frac{ 3-x }{1+x}, & x \geq 1. \end{cases}$	[-1,8; 1,8]
7.	$z = \begin{cases} \frac{1+x}{\sqrt[3]{1+x^2}}, & x \leq 0, \\ -x + 2e^{-2x}, & x \in (0,1), \\ 2-x ^{1/3}, & x \geq 1. \end{cases}$	[-2, 2]
8.	$z = \begin{cases} \frac{1+x}{1+x^2}, & x < 0, \\ \sqrt{1 + \frac{x}{1+x}}, & x \in [0,1), \\ 2 \sin(3x) , & x \geq 1. \end{cases}$	[-2; 1,8]
9.	$z = \begin{cases} \frac{1+x+x^2}{1+x^2}, & x < 0, \\ \sqrt{1 + \frac{2x}{1+x^2}}, & x \in [0,1), \\ 2 0,5 + \sin(x) , & x \geq 1. \end{cases}$	[-1,7; 1,5]
10.	$z = \begin{cases} 1 + \frac{3+x}{1+x^2}, & x < 0, \\ \sqrt{1+(1-x)^2}, & x \in [0,1), \\ \frac{1+x}{1+\cos^2(x)}, & x \geq 1. \end{cases}$	[-1,5; 1,8]
11.	$z = \begin{cases} \frac{1+2x}{1+x^2}, & x < 0, \\ \sin^2(x)\sqrt{1+x}, & x \in [0,1), \\ \sin^2(x)e^{0,2x}, & x \geq 1. \end{cases}$	[-1,4; 1,9]

12.	$z = \begin{cases} \frac{ x }{1+x^2} e^{-2x}, & x < 0, \\ \sqrt{1+x^2}, & x \in [0, 1), \\ \frac{1+\sin(x)}{1+x} + 3x, & x \geq 1. \end{cases}$	[-1,4; 1,4]
13.	$y = \begin{cases} 0,9 \cdot \lg x + \sqrt[3]{ x }, & x \geq 1,2 \\ \cos x + \sin(x^2), & 1,2 > x > 0,9 \\ 1,9 \cos x + 3x^2, & x \leq 0,9 \end{cases}$	[0,8; 2]
14.	$y = \begin{cases} -0,5t^{2\ln t}, & 1 \leq t \leq 2 \\ 1, & t < 1 \\ e^{-0,5t} \cos 2t, & t > 2 \end{cases}$	[0; 3]
15.	$y = \begin{cases} \pi x^2 - 7/x^2, & x < 1,3 \\ ax^3 + 7\sqrt{x}, & x = 1,3 \\ \lg(x + 7\sqrt{x}), & x > 1,3 \end{cases}$	[0,8; 2]
16.	$y = \begin{cases} 2,8x^2 - 0,3x + 4, & x < 1 \\ 2,8/x + \sqrt{x^2 + 1}, & 1 \leq x \leq 1,8 \\ (2,8 - 0,3x)/\sqrt{x^2 + 1}, & x > 1,8 \end{cases}$	[1; 2]
17.	$y = \begin{cases} \pi x^2 - 7/x^2, & x < 1 \\ 1,6x^3 + 7\sqrt{x}, & 1 \leq x \leq 1,4 \\ \ln(x + 7\sqrt{ x+1,6 }), & x > 1,4 \end{cases}$	[0,7; 2]
18.	$y = \begin{cases} 1,5 \cos^2 x, & x \leq 5 \\ 1,8 \cos x, & 5 < x < 7,1 \\ (x-2)^2 + 6, & x \geq 7,1 \end{cases}$	[0,2; 8]
19.	$y = \begin{cases} x\sqrt[3]{x-2,5}, & x > 3 \\ x \sin 2,5x, & 2 \leq x \leq 3 \\ e^{-2,5x} \cos 2,5x, & x < 2 \end{cases}$	[1; 5]

20.	$y = \begin{cases} 1,5x - \sin(1,5x), & x < 0,5 \\ 5 \sin^2 x + \lg(1,5x), & 0,5 \leq x \leq 0,8 \\ 1,5x^2 + \lg 1,5x, & x > 0,8 \end{cases}$	[0,1; 1]
-----	---	----------

Задание 2. В следующих задачах организовать выполнение алгоритма следующим образом: выбор вычисления по той или иной формуле – с помощью элемента управления переключатель, форматный или бесформатный вывод – с помощью элемента управления флажок.

№ вар.	Формулировка задачи
1.	Вычислить площадь и периметр прямоугольника, если задана длина одной стороны (a) и коэффициент n (%), позволяющий вычислить длину второй стороны ($b=n*a$).
2.	Вычислить периметр и площадь прямоугольного треугольника по заданным длинам двух катетов a и b .
3.	Дана длина ребра куба. Найти площадь грани, площадь полной поверхности и объем этого куба.
4.	Вычислить площади геометрических фигур: прямоугольника и треугольника по заданным сторонам.
5.	Вычислить площади геометрических фигур: трапеции и круга. $S_T = (a + b)h / 2$
6.	По известному радиусу вычислить объем и площадь поверхности шара. $V = \frac{4}{3}\pi r^3, S = 4\pi r^2$
7.	Заданы координаты трех вершин треугольника. Найти его периметр и площадь.
8.	Даны два числа. Найти среднее арифметическое кубов этих чисел и среднее геометрическое модулей этих чисел.
9.	Даны два числа. Вычислить их сумму, разность, произведение и частное.
10.	Дана сторона равностороннего треугольника. Найти площадь этого треугольника, его высоту, радиусы вписанной и описанной окружностей. $S = \frac{\sqrt{3}}{4}a^2, h = \frac{\sqrt{3}}{2}a, R = \frac{2}{\sqrt{3}}a, r = \frac{a}{2\sqrt{3}}$

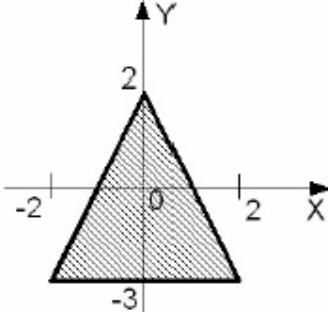
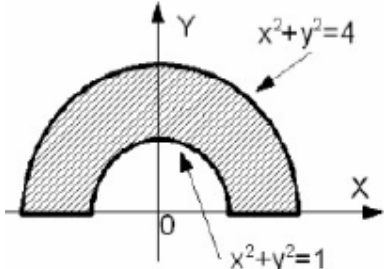
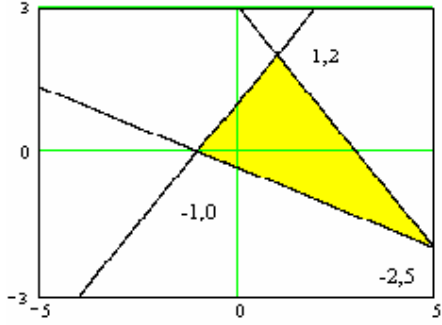
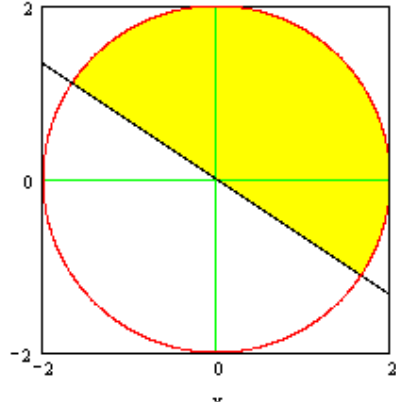
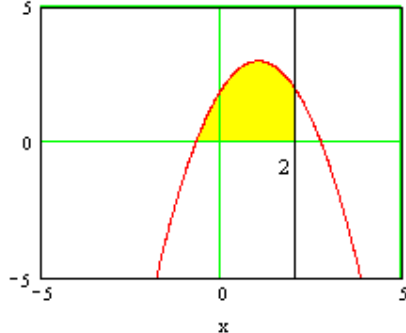
11.	Вычислить объем и площадь полной поверхности цилиндра, если известны высота и радиус основания. $V = \pi r^2 h, s = 2\pi r(r + h)$
12.	Вычислить значение функции для заданного аргумента для функции, изображенной на графике
13.	Заданы длина, ширина и высота параллелепипеда. Определить его объем и площадь поверхности.
14.	Для двух целых чисел A и B определить сумму S , разность R и среднее арифметическое Sr .
15.	В зависимости от дня недели, выдать сообщение о его номере. В зависимости от номера месяца, выдать его название.
16.	Напечатать числа a, b, c в порядке возрастания
17.	Найти количество отрицательных чисел среди a, b, c и абсолютное значение суммы этих чисел.
18.	Записать программу, которая на ввод времени суток выводит соответствующее пожелание доброго утра, доброго дня, доброго вечера и спокойной ночи.
19.	Определить результат гадания на ромашке – «любит – не любит», взяв за исходное данное количество лепестков n .
20.	Даны действительные числа x и y , не равные друг другу. Меньшее из этих двух чисел заменить половиной их суммы, а большее – их удвоенным произведением.

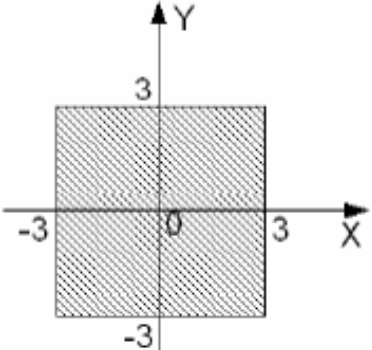
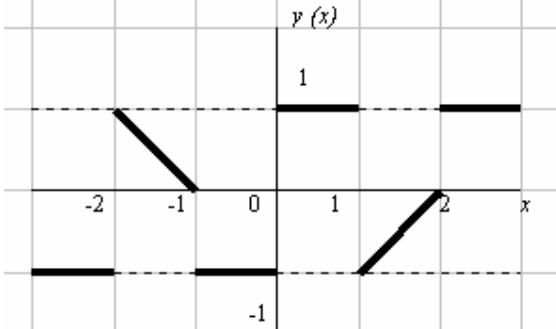
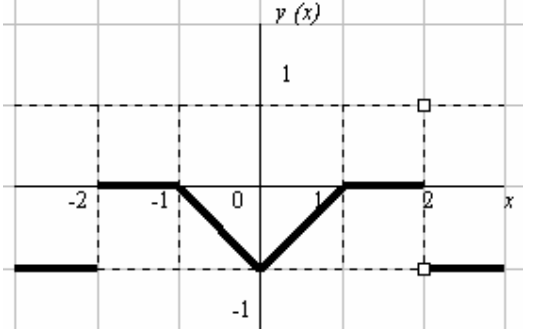
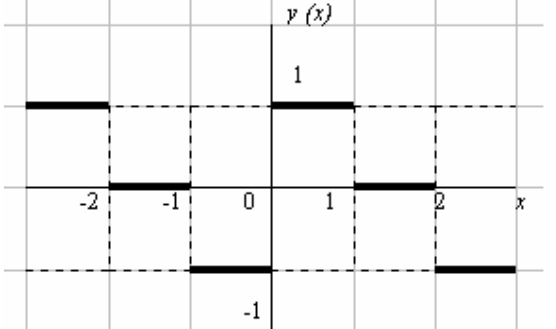
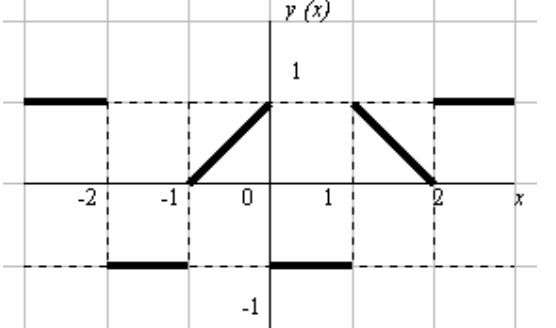
4.2. ВТОРОЙ УРОВЕНЬ СЛОЖНОСТИ

Организовать разветвляющийся процесс для решения следующих задач.

№ вар.	Формулировка задачи
1.	Даны две точки $A(x_1, y_1)$ и $B(x_2, y_2)$. Составить алгоритм, определяющий, которая из точек находится ближе к началу координат
2.	На плоскости XOY задана своими координатами точка A . Указать, где она расположена (на какой оси или в каком координатном угле)
3.	Заданы три стороны треугольника a, b и c . Определить является ли этот треугольник прямоугольным, и какая сторона служит гипотенузой
4.	Заданы радиус круга R и сторона квадрата A . Определить, можно ли вписать квадрат в круг

5.	Вычислить значение функции, изображенной на графике, для заданного значения аргумента	
6.	Вычислить значение функции, изображенной на графике, для заданного значения аргумента	
7.	Вычислить значение функции, изображенной на графике, для заданного значения аргумента	
8.	Определить принадлежит ли некоторая точка М с произвольными координатами x, y закрашенной области.	
9.	Определить принадлежит ли некоторая точка М с произвольными координатами x, y закрашенной области.	

10.	<p>Определить принадлежит ли некоторая точка М с произвольными координатами x, y закрашенной области.</p>	
11.	<p>Определить принадлежит ли некоторая точка М с произвольными координатами x, y закрашенной области.</p>	
12.	<p>Определить принадлежит ли некоторая точка М с произвольными координатами x, y закрашенной области.</p>	
13.	<p>Определить принадлежит ли некоторая точка М с произвольными координатами x, y закрашенной области. Уравнение прямой $y = -\frac{2}{3}x$</p>	
14.	<p>Определить принадлежит ли некоторая точка М с произвольными координатами x, y закрашенной области. Уравнение параболы $y = 2 + 2x - x^2$</p>	

15.	<p>Определить принадлежит ли некоторая точка М с произвольными координатами x, y закрашенной области.</p>	
16.	<p>Записать программу, которая на ввод значения аргумента выдает значение функции, заданной графиком.</p>	
17.	<p>Записать программу, которая на ввод значения аргумента выдает значение функции, заданной графиком.</p>	
18.	<p>Записать программу, которая на ввод значения аргумента выдает значение функции, заданной графиком.</p>	
19.	<p>Записать программу, которая на ввод значения аргумента выдает значение функции, заданной графиком.</p>	

20.	<p>Записать программу, которая на ввод значения аргумента выдает значение функции, заданной графиком.</p>	
-----	---	--

4.3. ТРЕТИЙ УРОВЕНЬ СЛОЖНОСТИ

Организовать разветвляющийся процесс для решения следующих задач.

№ вар.	Формулировка задачи
1.	Пройдет ли кирпич со сторонами x , y , z сквозь прямоугольное отверстие со сторонами r и s . Стороны отверстия должны быть параллельны граням кирпича.
2.	Можно ли коробку размером $a \times b \times c$ упаковать в посылку размером $r \times s \times t$? «Углом» укладывать нельзя.
3.	Можно ли из круглой заготовки радиуса r вырезать две прямоугольные пластинки с размерами $a \times b$ и $c \times d$?
4.	Лежит ли точка $M(x_m, y_m)$ внутри треугольника, заданного координатами своих вершин $A(x_A, y_A)$, $B(x_B, y_B)$, $C(x_C, y_C)$?
5.	Путник двигался t_1 часов со скоростью v_1 , а затем t_2 часов – со скоростью v_2 и t_3 часов – со скоростью v_3 . За какое время он одолел первую половину пути, после чего запланировал привал?
6.	Проверить лежит ли окружность $(x - a_1)^2 + (y - b_1)^2 = r_1^2$ целиком внутри окружности $(x - a_2)^2 + (y - b_2)^2 = r_2^2$ и наоборот.
7.	Можно ли на прямоугольном участке застройки размером a на b метров разместить два дома размером p на q и r на s метров? Дома можно располагать только параллельно сторонам участка.
8.	Дано число x . Вывести в порядке возрастания числа: $\sin x$, $\cos x$, $\ln x$. Если при каком либо x некоторые из выражений не имеют смысла, вывести сообщение об этом и сравнивать значения только тех, которые имеют смысл.

9.	Даны три положительных числа. Определить, можно ли построить треугольник с длинами сторон, равным этим числам. Если можно, то ответить на вопрос, является ли он остроугольным.
10.	По заданным трем числам определить, является ли сумма каких-либо из них положительной.
11.	Даны действительные числа a, b, c . Удвоить эти числа, если $a < b < c$, и заменить их абсолютными значениями, если это не так.
12.	Осуществить перевод величин из радианной меры в градусную и наоборот. Программа должна запрашивать, какой перевод нужно осуществить, и выполнять указанное действие.
13.	Определить будут ли прямые $A_1x+B_1y+C_1$ и $A_2x+B_2y+C_2$ перпендикулярны. Если нет, то найти угол между ними.
14.	Известно, что из четырех чисел a_1, a_2, a_3 и a_4 одно отлично от трех других, равных между собой; присвоить номер этого числа переменной n .
15.	Даны три точки $A(x_1, y_1), B(x_2, y_2)$ и $C(x_3, y_3)$. Определить, будут ли они расположены на одной прямой. Если нет, то вычислить угол $\angle ABC$.
16.	Написать программу решения уравнения $ax^3+bx=0$ для произвольных a, b .
17.	Даны три положительных числа a, b, c . Проверить, могут ли они быть длинами сторон треугольника. Если да, то вычислить его площадь.
18.	В доме M этажей и всего один подъезд; на каждом этаже по 3 квартиры; лифт может останавливаться только на нечетных этажах. Человек садится в лифт и набирает номер нужной ему квартиры N . На каком этаже остановится лифт?
19.	Написать программу решения системы линейных уравнений $\begin{cases} a_1x + b_1y = c_1 \\ a_2x + b_2y = c_2 \end{cases}$
20.	Проверить, не приводит ли к переполнению суммирование двух целых чисел A и B (т.е. к результату большему, чем 32767). Если переполнение, сообщить об этом, если нет – вывести сумму.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. С помощью какого оператора реализуется алгоритмическая структура «Развилка»? Нарисуйте ее блок-схему.
2. С помощью какого оператора реализуется алгоритмическая структура «Выбор»? Нарисуйте ее блок-схему.
3. Перечислите знаки отношений
4. Перечислите основные логические операции.
5. Когда применяется условный оператор?
6. Назовите два вида условного оператора.
7. Что позволяет составной оператор?
8. Что позволяет делать оператор выбора?
9. Что такое селектор?
10. Приведите пример на использование оператора условия.
11. Приведите пример на использование оператора выбора.
12. Для чего служит кнопка с независимой фиксацией?
13. Опишите свойство `Checked`.
14. Изучите и опишите информацию о компоненте `RadioButton` палитры `Standard` по справке `Delphi`.
15. В чем различие между объектами `CheckBox` и `RadioButton`?
16. Назначение объекта `RadioGroup`.
17. Как задать список элементов-переключателей в панели `RadioGroup`?
18. По какому свойству `RadioGroup` определяется выбранный переключатель? Как задать этому свойству значение.
19. Какое свойство объекта `Label` можно использовать для вывода ответа?
20. Какое свойство объекта `Edit` можно использовать для вывода ответа?