

Федеральное агентство по образованию

Государственное образовательное учреждение
высшего профессионального образования
«СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ИНДУСТРИАЛЬНЫЙ
УНИВЕРСИТЕТ»

Кафедра прикладной информатики

ПРОГРАММИРОВАНИЕ ЦИКЛИЧЕСКИХ АЛГОРИТМОВ В ОБЪЕКТ PASCAL

*Рекомендации к выполнению лабораторной работы по
дисциплине «Информатика»*

Специальности:

Городское строительство и хозяйство (270105)

Производство строительных материалов, изделий и конструкций
(270106)

Теплогазоснабжение и вентиляция (270109)

Водоснабжение и водоотведение (270112)

Проектирование зданий (270114)

Экспертиза и управление недвижимостью (270115)

Новокузнецк 2010

УДК 004.43(07)
П784

Рецензент:
Кафедра информационных технологий в металлургии
ГОУ ВПО СибГИУ
(Зав. кафедрой д.т.н., проф. В.П. Цымбал)

П784 Программирование циклических алгоритмов в Object Pascal: метод. указ. / Сост. Л.Д. Павлова, Д.Н. Климова: СибГИУ. – Новокузнецк, 2010. – 21 с.

Описаны операторы Object Pascal, предназначенные для организации циклических вычислительных процессов, рассмотрены примеры программ, приведены задания для самостоятельной работы.

Предназначена для студентов специальностей: 270105 Городское строительство и хозяйство, 270106 Производство строительных материалов, изделий и конструкций, 270109 Теплогазоснабжение и вентиляция, 270112 Водоснабжение и водоотведение, 270114 Проектирование зданий, 270115 Экспертиза и управление недвижимостью, а также может быть рекомендована для студентов других специальностей.

ЦЕЛЬ РАБОТЫ

Изучить операторы *Object Pascal*, которые используются для организации циклов, приобрести навыки программирования задач с циклическими алгоритмами.

ОБЩИЕ СВЕДЕНИЯ

Циклический алгоритм - это алгоритм, в котором определенная группа действий выполняется некоторое количество раз.

Переменная, изменяющаяся с некоторым шагом, называется *переменной цикла*. Переменная цикла определяется своим начальным значением, конечным значением и шагом изменения. Например, если переменная цикла изменяется следующим образом:

$$\varphi = 10^\circ, 20^\circ, 30^\circ, 40^\circ, 50^\circ, 60^\circ, 70^\circ, 80^\circ, 90^\circ,$$

тогда она однозначно определяется своим начальным значением $\varphi_n = 10^\circ$, конечным значением $\varphi_k = 90^\circ$ и шагом $h = 10^\circ$.

Повторяющаяся группа действий называется *телом цикла*. Количество повторений тела цикла определяется *условием окончания цикла*.

Если условие окончания цикла расположено перед телом цикла, то такой цикл называется *циклом с предусловием*.

Если условие окончания цикла расположено после тела цикла, то такой цикл называется *циклом с постусловием*.

Если количество повторений тела цикла определено по условию задачи, то такой цикл называется *циклом с параметром*.

Для организации циклических вычислительных процессов в языке *Object Pascal* Паскаль используются оператор цикла с предусловием, оператор цикла с постусловием и оператор цикла с параметром.

СОСТАВНОЙ ОПЕРАТОР

Составной оператор - это объединение нескольких операторов в одну операторную группу. Группа состоит из произвольного числа операторов, ограниченных операторными скобками **Begin** и **End**.

Формат составного оператора:

Begin

<оператор>;

<оператор>;

. . .

<оператор>;

End;

Составной оператор воспринимается как единое целое и может находиться в любом месте программы, где синтаксически допустимо наличие оператора.

Составной оператор используется в операторах условия для определения группы действий, которые необходимо выполнить в зависимости от условия.

Пример использования составного оператора в операторе условия:

If (x >= a) and (x <= b) **then**

Begin

Sum := Sum + x;

Kol := Kol+1;

End;

Составной оператор используется в операторах цикла для определения циклического участка программы (тела цикла).

Пример использования составного оператора в операторе цикла с параметром:

For i := 1 **To** n **Do**

Begin

Readln(a);

Sum := Sum + a;

End;

ОПЕРАТОР ЦИКЛА С ПРЕДУСЛОВИЕМ

Оператор цикла с предусловием используется в тех случаях, когда заранее неизвестно число повторений тела цикла. Этот оператор осуществляет проверку условия окончания цикла перед каждым выполнением тела цикла.

Формат оператора цикла с предусловием:

While <выражение> **Do** <оператор >;

где <выражение> - выражение логического типа, которое может принимать значение *True* или *False*;

<оператор> - простой или составной оператор.

Оператор, расположенный после зарезервированного слова **Do**, образует тело цикла. Если тело цикла состоит из двух или более операторов, то они **обязательно** должны заключаться в операторные скобки **Begin** и **End**, образуя составной оператор.

Выполнение оператора цикла с предусловием:

1. Перед каждым выполнением тела цикла вычисляется значение логического выражения.
2. Если выражение принимает значение *True*, то выполняется тело цикла, затем происходит переход на начало цикла.
3. Если значение выражения равно *False*, происходит выход из цикла и переход к оператору программы, следующему за оператором цикла.
4. Если перед первым выполнением цикла значение выражения равно *False*, тело цикла не выполняется ни разу и происходит переход к оператору, следующему за оператором цикла.

В теле цикла обязательно должен быть оператор, изменяющий значение переменной цикла, которая определяет условие выхода из цикла, иначе цикл получится бесконечным.

Задача. Составить программу вычисления суммы нечетных чисел в интервале от 1 до 100, используя оператор цикла с предусловием.

{Программа с использованием оператора цикла с предусловием}

Program DemoWhile;

Var x, Sum: **Integer**;

Begin

Sum := 0;

x := 1;

{Цикл с предусловием}

While x <= 100 **do**

{Начало тела цикла}

Begin

Sum := Sum + x;

{Изменение переменной цикла на величину шага}

x := x + 2;

End;

{Конец тела цикла}

Writeln (' Сумма нечетных чисел равна: ', Sum:4);

End.

Результат выполнения программы

Сумма нечетных чисел равна: 2500

ОПЕРАТОР ЦИКЛА С ПОСТУСЛОВИЕМ

Оператор цикла с постусловием используется в тех случаях, когда заранее неизвестно число повторений цикла. Этот оператор осуществляет проверку условия окончания цикла после первого выполнения тела цикла.

Формат оператора цикла с постусловием:

Repeat

<оператор>;

. . .

<оператор>;

Until <выражение>;

где <выражение> - выражение логического типа, которое может принимать значение *True* или *False*;

<оператор> - простой или составной оператор.

Операторы, расположенные между зарезервированными словами **Repeat** и **Until**, образуют тело цикла.

Выполнение оператора цикла с постусловием:

1. Выполняется тело цикла.
2. Вычисляется значение логического выражения.

3. Если значение выражения равно *False*, то тело цикла выполняется еще раз.

4. Если значение выражения равно *True*, то происходит выход из цикла и переход к оператору, следующему за оператором цикла.

В теле цикла обязательно должен быть оператор, изменяющий значение переменной цикла, которая определяет условие выхода из цикла, иначе цикл получится бесконечным.

Отличительные особенности оператора цикла с постусловием:

- оператор цикла с постусловием выполняется, по крайней мере, один раз, независимо от значения выражения;
- тело цикла выполняется, пока значение выражения равно *False*;
- в теле может находиться произвольное количество операторов без операторных скобок **Begin** и **End**.

Задача. Составить программу вычисления суммы четных чисел в интервале 1 до 100, используя оператор цикла с постусловием.

{Программа с использованием оператора цикла с постусловием}

```
Program DemoRepeat;  
Var x, Sum: Integer;  
Begin  
Sum := 0;  
x := 2;  
{Цикл с постусловием}  
Repeat  
{Начало тела цикла}  
Sum := Sum + x;  
{Изменение переменной цикла на величину шага}  
x := x + 2;  
{Конец тела цикла}  
Until x > 100;  
Writeln (' Сумма четных чисел равна: ', Sum:4);  
End.
```

Результат выполнения программы

Сумма четных чисел равна: 2550

ОПЕРАТОР ЦИКЛА С ПАРАМЕТРОМ

Оператор цикла с параметром используется в тех случаях, когда заранее известно сколько раз должна повториться циклическая часть программы. Этот оператор осуществляет проверку условия окончания цикла перед выполнением тела цикла.

Оператор цикла с параметром может быть представлен в двух форматах в зависимости от шага изменения переменной цикла.

Формат 1 оператора цикла с параметром:

For <переменная цикла> := <B1> **To** <B2> **Do** <оператор>;

Формат 2 оператора цикла с параметром:

For <переменная цикла>:= <B1> **DownTo** <B2> **Do**<оператор>;

где <переменная цикла> – переменная типа **Integer**, **Char**, **Boolean**;

<B1>, <B2> – выражения типа **Integer**, **Char**, **Boolean**;

B1 – начальное значение переменной цикла;

B2 – конечное значение переменной цикла;

<оператор> – простой или составной оператор.

Переменная цикла, выражение <B1>, выражение <B2> должны быть **обязательно** одного и того же типа.

Формат 1 предусматривает постоянный шаг изменения переменной цикла равный 1, при этом $B1 \leq B2$.

Формат 2 предусматривает постоянный шаг изменения переменной цикла равный -1, при этом $B1 \geq B2$.

Оператор, расположенный после зарезервированного слова **Do**, образует тело цикла. Если тело цикла состоит из двух или более операторов, то они **обязательно** должны заключаться в операторные скобки **Begin** и **End**, образуя составной оператор.

Выполнение оператора цикла с параметром

1. Перед первым выполнением тела цикла вычисляется значение арифметических выражений B1 и B2.

2. Переменной цикла присваивается значение выражения B1.

3. Если полученное значение переменной цикла не превышает конечного значения B2, то выполняется тело цикла.

4. Выполняется возврат на начало оператора цикла.
5. Автоматически значение переменной цикла увеличивается на величину шага.
6. Если полученное значение переменной цикла не превышает конечного значения В2, то выполняется тело цикла.
7. Если полученное значение переменной цикла превысило конечное значение В2, то выполняется переход к оператору, следующему за оператором цикла.
8. Если перед первым выполнением оператора, переменная цикла превышает конечное значение В2, то тело цикла не выполняется ни разу и происходит переход к следующему оператору программы.

В теле оператора цикла не допускается изменение переменной цикла, так как она изменяется автоматически только на величину ± 1 при каждом возврате на начало оператора цикла.

Задача. Составить программу вычисления суммы чисел в интервале 1 до 100, используя оператор цикла с параметром.

{Программа с использованием оператора цикла параметром}

```
Program DemoFor;  
Var x, Sum: Integer;  
Begin  
Sum := 0;  
{Цикл с параметром}  
For x := 1 to 100 do  
{Начало тела цикла}  
Begin  
Sum := Sum + x;  
End;  
{Конец тела цикла}  
Writeln (' Сумма чисел равна: ', Sum:4);  
End.
```

Результат выполнения программы

Сумма чисел равна: 5050

ПРИМЕРЫ ВЫПОЛНЕНИЯ ЗАДАНИЯ

Задание 1. Составить программу используя оператор цикла с предусловием.

Задача

Вычислить значения функций $\text{Sin}(\varphi)$ на интервале $[a,b]$ с заданным шагом.

Программа с использованием оператора цикла с предусловием

```
Program P1;  
Var a, b, h : Integer;  
    fi, fir, y : Real;  
Begin  
    Write(' Введите левую границу интервала a: ');  
    Readln(a);  
    Write(' Введите правую границу интервала b: ');  
    Readln(b);  
    Write(' Введите шаг h: ');  
    Readln(h);  
    fi := a;  
    While fi <= b Do  
    Begin  
        fir := fi*pi/180;  
        y := Sin(fir);  
        Writeln(' fi = ', fi:6:2, ' y = ', y:5:2);  
        fi := fi+h;  
    End;  
End.
```

Результат выполнения программы

```
Введите левую границу интервала a: 0  
Введите правую границу интервала b: 360  
Введите шаг h: 90  
fi = 0.00 y = 0.00  
fi = 90.00 y = 1.00  
fi = 180.00 y = 0.00  
fi = 270.00 y = -1.00  
fi = 360.00 y = 0.00
```

Задание 2. Составить программу используя оператор цикла с параметром.

Задача

Задана последовательность из n вещественных чисел. Определить величину максимального элемента последовательности.

Программа с использованием оператора цикла с параметром

```
Program P2;  
Var a, max: Real;  
i, n: Integer;  
Begin  
  Write(' Введите количество чисел: ');  
  Readln(n);  
  Write(' Введите первое число: ');  
  Readln(a);  
  max := a;  
  For i := 2 To n Do  
    Begin  
      Write(' Введите следующее число: ');  
      Readln(a);  
      If a > max then max := a;  
    End;  
  Writeln('Максимальный элемент = ', max:5:2);  
End.
```

Результат выполнения программы

```
Введите количество чисел: 5  
Введите первое число: 19  
Введите следующее число: 12  
Введите следующее число: 18  
Введите следующее число: 10  
Введите следующее число: 17  
Максимальный элемент = 18.00
```

Задание 3. Составить программу используя оператор цикла с постусловием.

Задача

Задано натуральное число. Определить количество цифр в записи числа.

Программа с использованием оператора цикла с постусловием


```
Program P3;  
Var b, del: Real;  
k: Integer;  
Begin  
Write(' Введите натуральное число: '),  
Readln(b);  
k := 0;  
Repeat  
del :=b/10;  
k := k + 1;  
b := del;  
Until del < 1;  
Writeln('Количество цифр в записи числа = ', k:3);  
End.
```

Результат выполнения программы

Введите натуральное число: 123
Количество цифр в записи числа = 3

РЕАЛИЗАЦИЯ ПРОГРАММЫ В КОНСОЛЬНОМ ПРИЛОЖЕНИИ DELPHI

Для создания консольного приложения в среде *Delphi* необходимо выполнить следующие действия.

1). Запустить среду *Delphi*, выполнив последовательность действий:  пуск → Все программы → Borland Delphi 7 → Delphi 7.

2). В горизонтальном меню **File** выполнить команду **New**, затем в раскрывшемся окне – команду **Other**.

Откроется диалоговое окно **New Items** (рисунок 1).

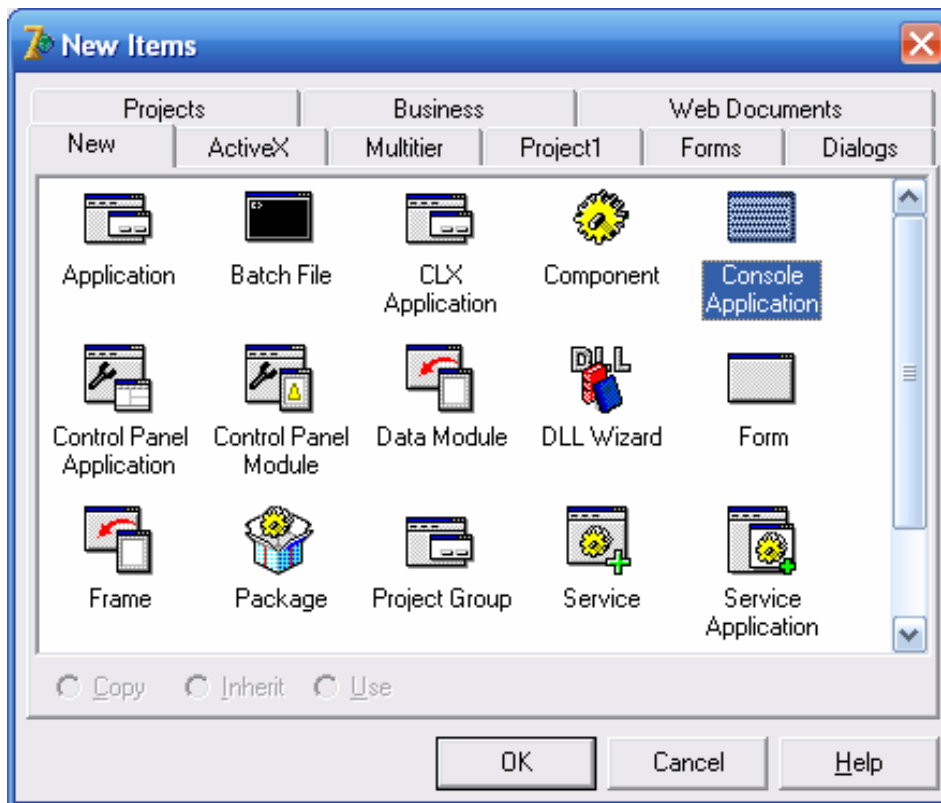


Рисунок 1 – Диалоговое окно **New Items**

3). На вкладке **New** диалогового окна **New Items** выбрать тип



Console Application.

создаваемого приложения

Откроется окно **Project 2.dpr**, в котором находится шаблон главной процедуры консольного приложения (рисунок 2).

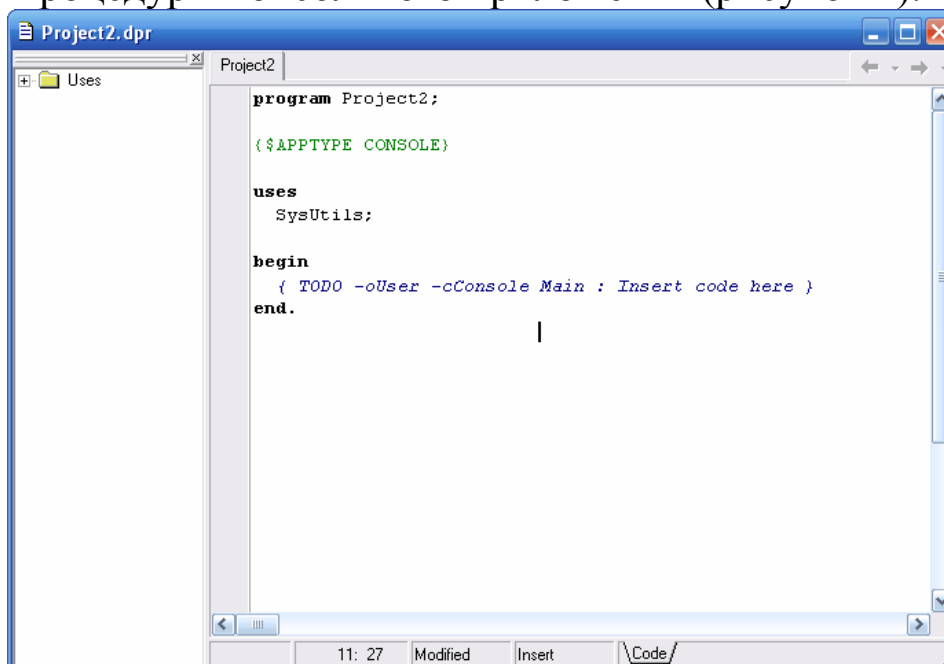


Рисунок 2 – Шаблон главной процедуры консольного приложения

4). В окне **Project2.dpr** необходимо выполнить следующие действия (рисунок 3):

– после строк

```
program Project2;  
  {$APPTYPE CONSOLE}
```

Uses

```
  SysUtils;
```

поместить раздел описания переменных:

```
Var b, del: Real;
```

```
k: Integer;
```

– между строками:

```
begin
```

```
  { TODO -oUser -cConsole Main : Insert code here }
```

```
end;
```

записать фрагмент кода программы:

```
Write('Vvedite naturalnoe chislo: '),
```

```
Readln(b);
```

```
k := 0;
```

```
Repeat
```

```
del := b/10;
```

```
k := k + 1;
```

```
b := del;
```

```
Until del < 1;
```

```
Writeln('Kolichestvo cifr v zapisi chisla = ', k:3);
```

```
Readln;
```

Замечание 1. В консольном приложении допускается использование в коде программы только символов английского алфавита.

Замечание 2. Для задержки окна консоли, чтобы посмотреть результаты выполнения программы, необходимо в конце программы добавить процедуру ожидания ввода **Readln**.

Без процедуры **Readln** окно консоли появится на экране монитора при выполнении программы и сразу же закроется после ее выполнения.

5). Для компиляции программы необходимо в горизонтальном меню **Project** выполнить команду **Compile Project2**.

Если компилятор обнаружит ошибку в программе, то в нижней части окна **Project2.dpr** появится сообщение об ошибке с указанием номера строки, в которой обнаружена ошибка.

В этом случае необходимо найти указанную строку, исправить ошибку и повторно выполнить компиляцию программы.

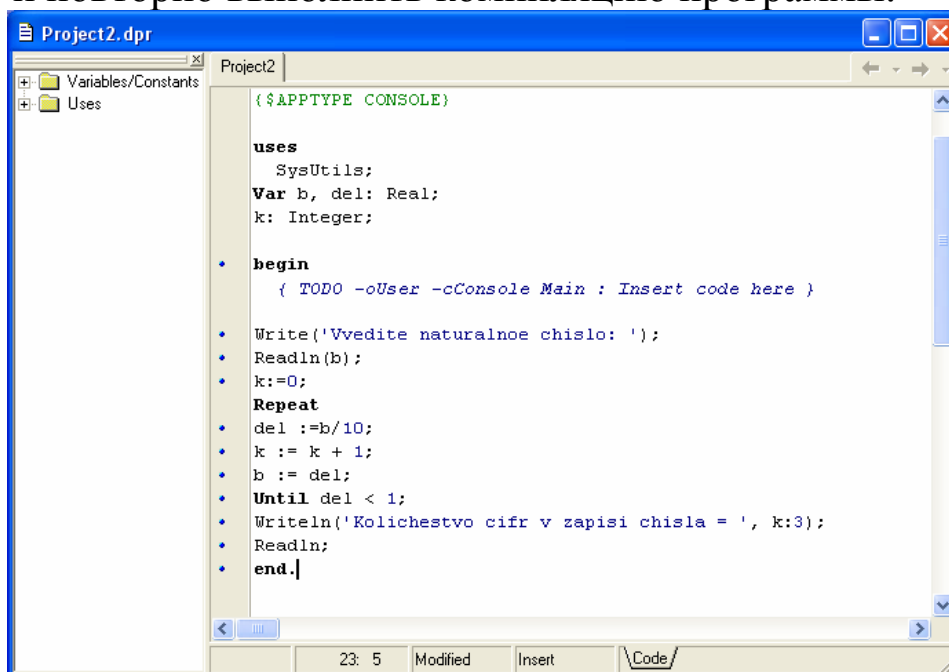


Рисунок 3 – Окно консольного приложения с кодом программы

6). Для запуска программы на выполнение необходимо в горизонтальном меню **Run** выполнить команду **Run**.

Откроется стандартное окно, в котором будет выполняться программа (рисунок 14).

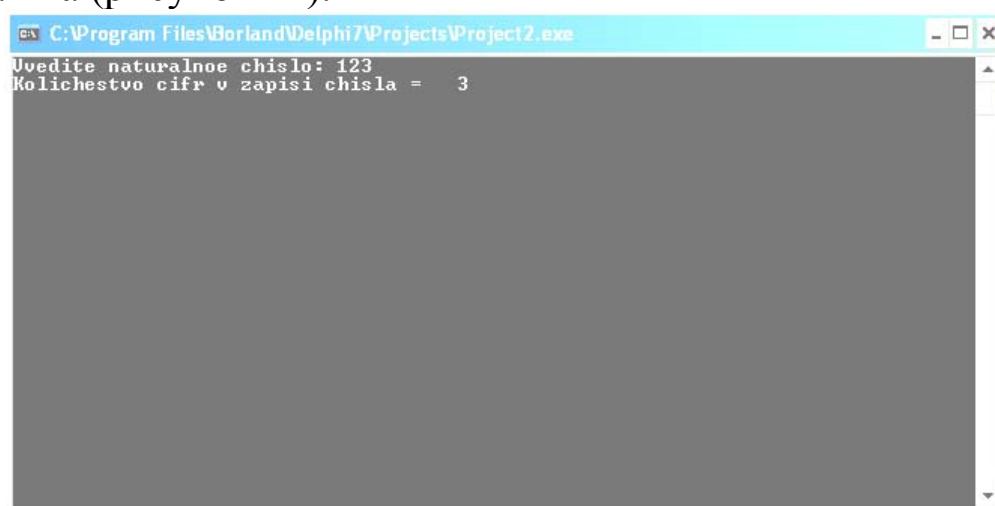


Рисунок 14 – Окно выполнения программы

7). Если при выполнении программы требуется ввести исходные данные, то по запросу программы необходимо

осуществить ввод с клавиатуры соответствующих значений (рисунок 14).

Замечание 3. При вводе нескольких значений их следует разделять пробелами.

Замечание 4. При вводе вещественного значения целую часть от дробной следует отделять точкой.

Замечание 5. В конце строки ввода необходимо нажать клавишу **Enter**.

8.) После завершения работы программы и просмотра результатов, окно консоли необходимо закрыть, нажав клавишу **Enter**.

9). Для сохранения программы в папку группы необходимо в горизонтальном меню **File** выполнить команду **Save as**.

Откроется диалоговое окно **Save Project2 as** (рисунок 15), в котором необходимо выполнить следующие действия:

- в раскрывающемся списке поля **Папка:** открыть папку группы;
- в поле **Имя файла:** ввести имя сохраняемого файла;
- щелкнуть по кнопке **Сохранить**.

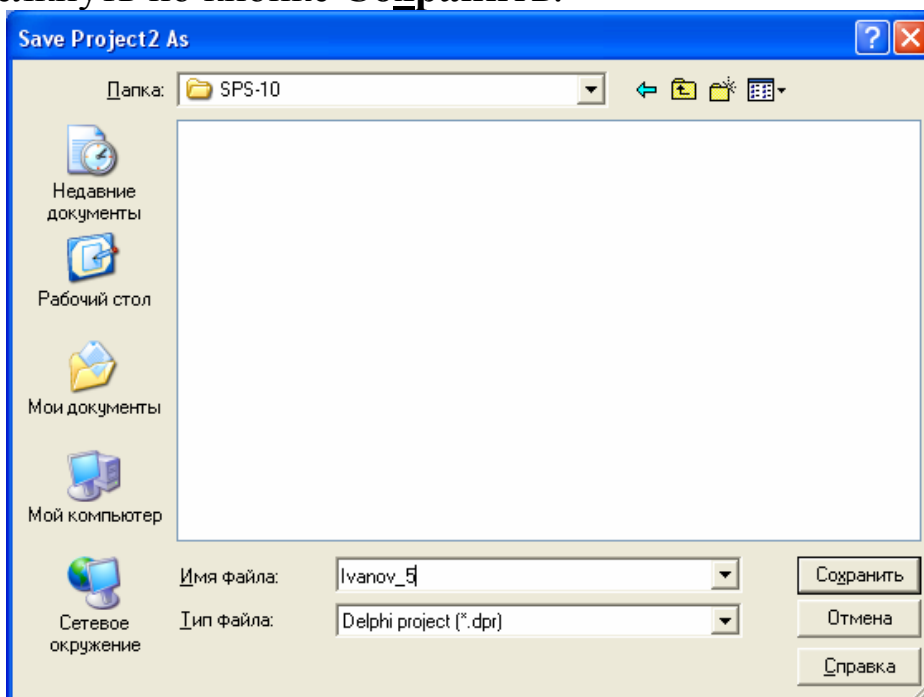


Рисунок 15 – Диалоговое окно Save Project2 As

10). Для завершения работы в среде *Delphi* необходимо в горизонтальном меню **File** выполнить команду **Exit**.

ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Задание 1. Составить программу используя оператор цикла с предусловием

Задача 1

Вычислить значения функций $\sin(\varphi)$ и $\cos(\varphi)$ на интервале $[a, b]$ с заданным шагом.

Задача 2

Вычислить количество отрицательных значений $\sin(\varphi)$ на интервале $[a, b]$ с заданным шагом.

Задача 3

Вычислить количество положительных значений $\cos(\varphi)$ на интервале $[a, b]$ с заданным шагом.

Задача 4

Вычислить сумму положительных значений $\sin(\varphi)$ на интервале $[a, b]$ с заданным шагом.

Задача 5

Вычислить сумму отрицательных значений $\cos(\varphi)$ на интервале $[a, b]$ с заданным шагом.

Задание 2. Составить программу используя оператор цикла с параметром

Задача 6

Задана последовательность из n вещественных чисел. Определить количество положительных и количество отрицательных элементов последовательности.

Задача 7

Задана последовательность из n вещественных чисел. Определить количество элементов последовательности, попадающих в заданный интервал $[a, b]$.

Задача 8

Задана последовательность из n вещественных чисел. Определить сумму и количество элементов последовательности, меньших заданного значения d .

Задача 9

Задана последовательность из n вещественных чисел. Определить количество четных и количество нечетных элементов последовательности.

Задача 10

Задана последовательность из n вещественных чисел. Определить сумму положительных элементов последовательности, не попадающих в заданный интервал $[a, b]$.

Задача 11

Задана последовательность из n целых чисел. Определить среднее арифметическое отрицательных элементов последовательности.

Задача 12

Задана последовательность из n целых чисел. Определить среднее арифметическое элементов последовательности, больших заданного значения d .

Задача 13

Задана последовательность из n вещественных чисел. Определить величину максимального из положительных элементов последовательности.

Задача 14

Задана последовательность из n вещественных чисел. Определить величину минимального из отрицательных элементов последовательности.

Задача 15

Задана последовательность из n вещественных чисел. Определить разность между максимальным и минимальным элементами последовательности.

Задача 16

Задана последовательность из n целых чисел. Определить порядковый номер минимального элемента последовательности.

Задача 17

Задана последовательность из n целых чисел. Определить порядковый номер максимального элемента последовательности.

Задача 18

Задана последовательность из n целых чисел. Определить, сколько из них принимает максимальное значение.

Задача 19

Задана последовательность из n вещественных чисел. Определить количество чисел, больших своих соседей, т. е. предыдущего и последующего числа.

Задача 20

Задана последовательность из n целых чисел. Определить, сколько раз в этой последовательности меняется знак.

Задание 3. Составить программу используя оператор цикла с постусловием

Задача 21

Задано целое положительное значение x . Определить сумму целых чисел, попадающих в интервал $[\sqrt{x}, x^2]$.

Задача 22

Задано целое неотрицательное число. Определить последнюю цифру в десятичной записи этого числа.

Задача 23

Задано целое неотрицательное число. Определить сумму четных цифр в десятичной записи этого числа.

Задача 24

Задано натуральное число. Определить количество нулей в десятичной записи этого числа.

Задача 25

Задано целое неотрицательное число. Определить количество нечетных цифр в десятичной записи этого числа.

Задача 26

Задано число в двоичной системе счисления. Определить количество нулей и количество единиц в записи этого числа.

Задача 27

Задано натуральное число. Определить количество делителей этого числа.

Задача 28

Задано натуральное число. Определить, является ли оно степенью числа 3 (например, число 27 является степенью числа 3, так как $27=3^3$, а число 30 – не является).

Задача 29

Задано целое положительное число $b>2$. Определить количество всех простых чисел из диапазона $[2, b]$.

Задача 30

Задано натуральное число. Определить, является ли оно совершенным, т. е. равным сумме всех своих делителей, кроме самого этого числа (например, число 6 совершенно: $6=1+2+3$).

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Какой алгоритм называется циклическим?
2. Что такое переменная цикла?
3. Какими параметрами характеризуется переменная цикла?
4. Что такое тело цикла?
5. Где может располагаться условие окончания цикла?
6. Какие операторы используются для организации циклов?
7. Что такое составной оператор?
8. Каков формат составного оператора?
9. Каков формат оператора цикла с предусловием?
10. Выражение какого типа используется в операторе цикла с предусловием?
11. Как выполняется оператор цикла с предусловием?
12. Каков формат оператора цикла с постусловием?
13. Выражение какого типа используется в операторе цикла с постусловием?
14. Как выполняется оператор цикла с постусловием?
15. Каковы отличительные особенности оператора цикла с постусловием?
16. В каких случаях может получиться бесконечный цикл?
17. Каковы форматы оператора цикла с параметром?
18. Выражения какого типа используются в операторе цикла с параметром?
19. Какова величина шага в операторе цикла с параметром?
20. Как выполняется оператор цикла с параметром?

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

1. Павлова Л.Д. Информатика. Учебный курс: учеб. пособие / Л.Д. Павлова, О.А. Кондратова, Н.В. Балицкая; СибГИУ. – Новокузнецк, 2009. - 320 с.
2. Острейковский В.А. Информатика : учеб. для вузов / В.А. Острейковский. – М. : Высшая школа, 2005. – 511 с.
3. Крылов Е.В. Техника разработки программ : в 2-х кн. Кн.1. Программирование на языке высокого уровня : учебник. – 3-е изд., перераб. и доп. / Е.В. Крылов, В.А. Острейковский, Н.Г. Типикин – М. : Высшая школа, 2007. – 375 с.
4. Фаронов В.В. Delphi. Программирование на языке высокого уровня: Учебник для вузов / В.В. Фаронов. – С-Пб.: Питер, 2003. – 640 с.
5. Фаронов В.В. Delphi 6 : учебный курс / В.В. Фаронов. – М. : Нолидж, 2001. – 672 с.
6. Галисеев Г.В. Программирование в среде Delphi 7. Самоучитель / Г.В. Галисеев. – М.: Вильямс, 2004. – 288с.

СОДЕРЖАНИЕ

ЦЕЛЬ РАБОТЫ	3
ОБЩИЕ СВЕДЕНИЯ	3
СОСТАВНОЙ ОПЕРАТОР	4
ОПЕРАТОР ЦИКЛА С ПРЕДУСЛОВИЕМ	5
ОПЕРАТОР ЦИКЛА С ПОСТУСЛОВИЕМ	6
ОПЕРАТОР ЦИКЛА С ПАРАМЕТРОМ	8
ПРИМЕРЫ ВЫПОЛНЕНИЯ ЗАДАНИЯ	10
РЕАЛИЗАЦИЯ ПРОГРАММЫ В КОНСОЛЬНОМ ПРИЛОЖЕНИИ DELPHI	12
ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ	17
ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ	20
РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА	21

Учебное издание

Составители

Павлова Лариса Дмитриевна
Климова Дарья Николаевна

**ПРОГРАММИРОВАНИЕ ЦИКЛИЧЕСКИХ АЛГОРИТМОВ
В ОБЪЕКТ PASCAL**

Рекомендации к выполнению лабораторной работы по дисциплине
«Информатика»

Технический редактор
Н.П. Лавренюк

Подписано в печать _____ 2010г. Формат бумаги 60 x 84
1/16. Бумага писчая. Печать офсетная. Усл. печ. л. ____ Уч. - изд. л.
_____ Тираж 100 экз. Заказ _____

Сибирский государственный индустриальный университет
654007, г. Новокузнецк, ул. Кирова, 42
Издательский центр СибГИУ