Министерство образования Республики Беларусь

Учреждение образования

БелорусскиЙ государственный университет

информатики и радиоэлектроники

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

|  |
| --- |
|  |
|  |

# оТЧЕТ

по лабораторной работе

на тему:

СЛожные ЦИКЛЫ

|  |  |  |
| --- | --- | --- |
| Выполнил  Студент гр. 951002 |  | В. Н. Протасеня |
| Проверил |  | Асс. Е.Е. Фадеева |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Минск, 2019

1. Теоретические сведения по теме лабораторной работы

**Алгоритм**

Алгоритм - система правил, четко описывающая последовательность действий, которые необходимо выполнить для решения задачи.

**Способы описания алгоритмa:**

1. запись на естественном языке (словесное описание)
2. изображение в виде схемы (графическое описание)
3. запись на алгоритмическом языке (составление программы)

**Свойства правильного алгоритма:**

1. Дискретность – значения величин в каждый следующий момент времени должны получаться по определенным правилам из значений величин, имевшихся в предшествующий момент времени.
2. Определенность (детерминированность) – каждое правило алгоритма должно быть однозначным. Значения величин, получаемых в какой-то момент времени, однозначно связаны со значениями величин, вычисленных ранее.
3. Результативность (конечность) – алгоритм должен приводить к решению задачи за конечное число шагов.
4. Массовость – алгоритм должен разрабатываться в общем виде так, чтобы его можно было применить для класса задач, различающихся лишь исходными данными.

**Типы алгоритмов:**

1. Линейные (четко прописанная цепочка действий, в которой нет условий)
2. Разветвляющиеся (разделение на несколько альтернативных ветвей алгоритма)
3. Циклические (многократное повторение одних и тех же операций над данными до удовлетворения нескольких условий)

**Циклы**

Циклический вычислительный процесс – это процесс, в котором отдельные участки вычислений выполняются многократно. Участок схемы, многократно повторяемый̆ в ходе вычислений, называется циклом. При повторениях обычно используются новые значения исходных данных.

**Виды циклов:**

В соответствии со взаимным расположением циклов в теле программы или алгоритма различают сле-

дующие циклы:

1. простые – циклы, не содержащие внутри себя других циклов;
2. сложные – циклы, содержащие внутри себя другие циклы;
3. вложенные (внутренние) – циклы, входящие в состав других циклов (цикл в цикле);
4. внешние – циклы, не являющиеся составной частью других циклов, но содержащие в своем составе внутренние циклы.

В зависимости от местоположения условия выполнения цикла различают следующие циклы:

* + 1. Циклы с предусловием
    2. Циклы с постусловием

В соответствии с видом условия выполнения циклы делятся на следующие виды:

* 1. циклы с параметром;
  2. итерационные циклы.

1. Задание №1 на лабораторную работу

Из пункта приведённой̆ ниже таблицы вычислить значение функции *f(x,n)* для *n = 10; 11; . . . 15* и значения *x*, изменяющегося от *xн = 0.6* до *xк = 1.1* с шагом ∆*x = 0.25*.

|  |  |
| --- | --- |
| 26 |  |

* 1. Постановка задачи

Задача: написать программу, которая вычисляет значения функции *f(x,n)* для *n = 10; 11; . . . 15* и значения *x*, изменяющегося от *xн = 0.6* до *xк = 1.1* с шагом ∆*x = 0.25* на языке Delphi в среде разработки Embarcadero Delphi.

* 1. Эскиз ожидаемого результата

n=10 x=0.60 y= *f(x,n)*

n=11 x=0.60 y= *f(x,n)*

n=12 x=0.60 y= *f(x,n)*

n=13 x=0.60 y= *f(x,n)*

n=14 x=0.60 y= *f(x,n)*

n=15 x=0.60 y= *f(x,n)*

n=10 x=0.85 y= *f(x,n)*

n=11 x=0.85 y= *f(x,n)*

n=12 x=0.85 y= *f(x,n)*

n=13 x=0.85 y= *f(x,n)*

n=14 x=0.85 y= *f(x,n)*

n=15 x=0.85 y= *f(x,n)*

n=10 x=1.10 y= *f(x,n)*

n=11 x=1.10 y= *f(x,n)*

n=12 x=1.10 y= *f(x,n)*

n=13 x=1.10 y= *f(x,n)*

n=14 x=1.10 y= *f(x,n)*

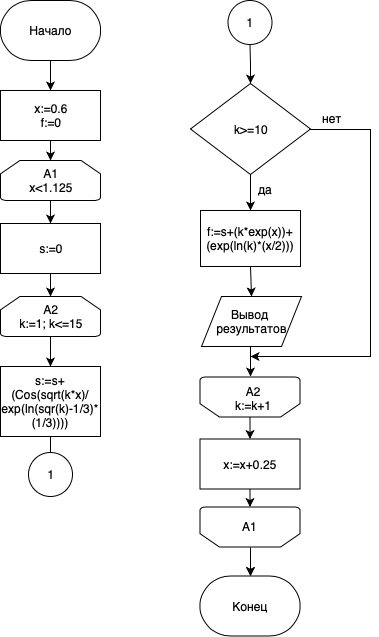
n=15 x=1.10 y= *f(x,n)*

1. Выполнение
   1. Разработка алгоритма

Таблица 3.1 используемые идентификаторы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Имя идентификатора | назначение | Тип идентификатора | Начальное значение | Закон изменения | Имя цикла, в котором происходит изменение переменной |
| x | Начальное значение аргумента | real | 0.6 | x:=x+0.25 | A1 |
| k | Количество элементов сигмы | integer | 1 | k:=k+1 | A2 |
| f | Значение функции | real | 0 |  | A2 |
| s | Сигма | real | 0 | s:=s+(Cos(sqrt(k\*x)/exp(ln(sqr(k)-1/3)\*(1/3)))) | A2 |

Рисунок 3.1 – Схема работы программы



Описание алгоритма:

Программу вычисляет значения функции *f(x,n)* для *n = 10; 11; . . . 15* и значения *x*, изменяющегося от *xн = 0.6* до *xк = 1.1* с шагом ∆*x = 0.25*. Программа состоит из 2 циклов с предусловием с параметром, где 1 цикл – внешний, сложный, 2 цикл – простой, внутренний.

* 1. Текст программы и его описание

program laba2part1;

{$APPTYPE CONSOLE}

{$R \*.res}

uses

System.SysUtils;

var x,f,s: real;

k: integer;

begin

x:=0.6;

f:=0;

while x<(1.125) do

begin

s:=0;

for k := 1 to 15 do

begin

s:=s+(Cos(sqrt(k\*x)/exp(ln(sqr(k)-1/3)\*(1/3))));

if k>=10 then

begin

f:=s+(k\*exp(x))+(exp(ln(k)\*(x/2)));

writeln('n=',k:2,' x=', x:4:2, ' f=', f:6:2);

end;

end;

x:=x+0.25;

end;

readln

end.

Описание программы: программа выводит значения функции *f(x,n)* для *n = 10; 11; . . . 15* и значения *x*, изменяющегося от *xн = 0.6* до *xк = 1.1* с шагом ∆*x = 0.25* для заданной в условии задания функции.

* 1. Тестирование и отладка программы

Таблица 3.2 Прохождение тестов программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Специфика тестирования | Номер теста | Вводимые данные | Ожидаемый результат | Полученный результат |
|  | 1 | –– | n=10 x=0.60 f= 28.30  n=11 x=0.60 f= 31.05  n=12 x=0.60 f= 33.80  n=13 x=0.60 f= 36.55  n=14 x=0.60 f= 39.30  n=15 x=0.60 f= 42.05  n=10 x=0.85 f= 33.39  n=11 x=0.85 f= 36.65  n=12 x=0.85 f= 39.92  n=13 x=0.85 f= 43.18  n=14 x=0.85 f= 46.44  n=15 x=0.85 f= 49.71  n=10 x=1.10 f= 40.19  n=11 x=1.10 f= 44.15  n=12 x=1.10 f= 48.11  n=13 x=1.10 f= 52.06  n=14 x=1.10 f= 56.02  n=15 x=1.10 f= 59.97 | Тест пройден |

Анализ прохождения тестов, причины возникших ошибок, внесение изменений в алгоритм и текст программы

Программа прошла тестирование, ошибок не обнаружено.

* 1. Итоговый текст программы

program laba2part1;

{$APPTYPE CONSOLE}

{$R \*.res}

uses

System.SysUtils;

var x,y,s: real;

k: integer;

begin

x:=0.6;

y:=0;

while x<(1.125) do

begin

s:=0;

for k := 1 to 15 do

begin

s:=s+(Cos(sqrt(k\*x)/exp(ln(sqr(k)-1/3)\*(1/3))));

if k>=10 then

begin

y:=s+(k\*exp(x))+(exp(ln(k)\*(x/2)));

writeln('n=',k:2,' x=', x:4:2, ' y=', y:6:2);

end;

end;

x:=x+0.25;

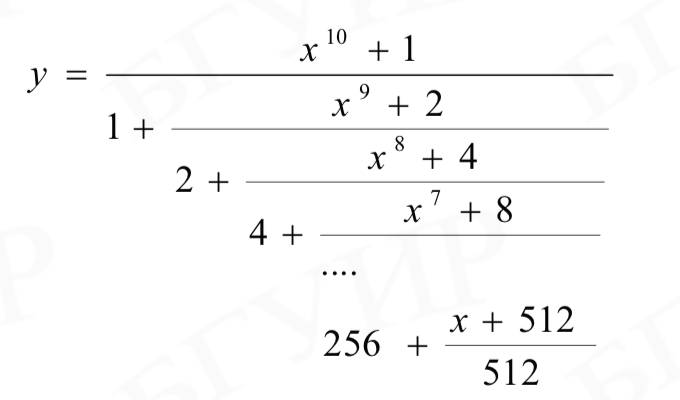
end;

readln

end.

1. Задание №2 на лабораторную работу

Вычислить цепную дробь для *X* = 0.5(0.05)0.8:



* 1. Постановка задачи

Задача: написать программу, которая вычисляет значение цепной дроби для значения *x*, изменяющегося от *xн = 0.5* до *xк = 0.8* с шагом ∆*x = 0.05* на языке Delphi в среде разработки Embarcadero Delphi.

* 1. Эскиз ожидаемого результата

x= 0.55 y(x)= значение цепной дроби

x= 0.60 y(x)= значение цепной дроби

x= 0.70 y(x)= значение цепной дроби

x= 0.75 y(x)= значение цепной дроби

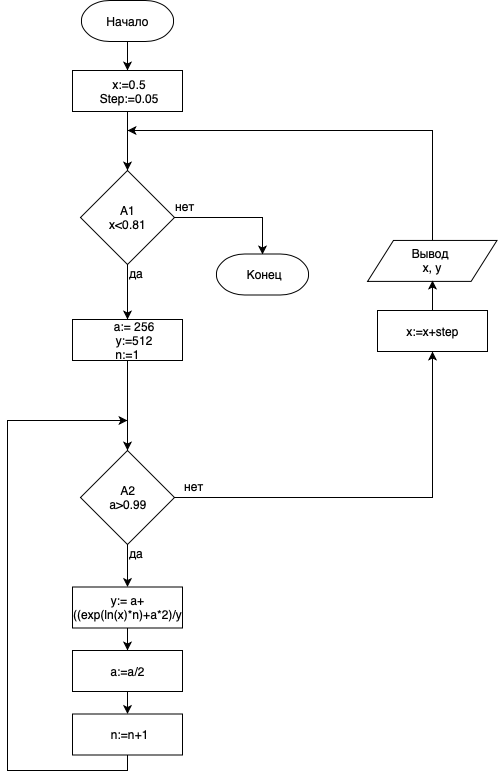
x= 0.80 y(x)= значение цепной дроби

1. Выполнение
   1. Разработка алгоритма

Таблица 5.1 используемые идентификаторы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Имя идентификатора | назначение | Тип идентификатора | Начальное значение | Закон изменения | Имя цикла, в котором происходит изменение переменной |
| x | Значение аргумента | real | 0.5 | x:=x+step | A1 |
| y | Значение функции | real | — |  | A2 |
| step | Шаг функции | real | 0.05 | Константа | — |
| a | Закономерность изменения числа в цепной дроби | real | 256 | a:=a/2 | A2 |
| n | Степень аргумента | integer | 1 | n:=n+1 | A2 |

Рисунок 6.1 – Схема работы программы



Описание алгоритма:

Программа вычисляет значение цепной дроби для значения *x*, изменяющегося от *xн = 0.5* до *xк = 0.8* с шагом ∆*x = 0.05*. Программа состоит из 2 циклов с предусловием с параметром, где 1 цикл – внешний, сложный, 2 цикл – простой, внутренний.

6.2 Текст программы и его описание

Program laba2part2;

{$R \*.res}

uses

System.SysUtils;

var x,y,Step,a:real;

n: integer;

begin

x:=0.5;

Step:=0.05;

while x<0.81 do

begin

a:= 256;

y:=512;

n:=1;

while a>0.99 do

begin

y:= a+ ((exp(ln(x)\*n)+a\*2)/y) ;

a:=a/2;

n:=n+1;

end;

x:=x+step;

y:= (exp(ln(x)\*n)+a\*2)/y ;

writeln('x= ',x:4:2, ' y= ',y:10:6);

end;

readln;

end.

Описание программы: программа выводит значение цепной дроби x от *xн = 0.5* (начальное значение аргумента x) до *xк = 0.8* (конечное значение аргумента x) с шагом ∆*x = 0.05* приращения аргумента для заданной в условии задания дроби.

* 1. Тестирование и отладка программы

Таблица 3.2 Прохождение тестов программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Специфика тестирования | Номер теста | Вводимые данные | Ожидаемый результат | Полученный результат |
|  | 1 | –– | x= 0.50 y= 0.585215  x= 0.55 y= 0.585871  x= 0.60 y= 0.587388  x= 0.65 y= 0.590654  x= 0.70 y= 0.597265  x= 0.75 y= 0.609926  x= 0.80 y= 0.632984 | Тест пройден |

Анализ прохождения тестов, причины возникших ошибок, внесение изменений в алгоритм и текст программы

Программа прошла тестирование, ошибок не обнаружено.

* 1. Итоговый текст программы

Program laba2part2;

{$R \*.res}

uses

System.SysUtils;

var x,y,Step,a:real;

n: integer;

begin

x:=0.5;

Step:=0.05;

while x<0.81 do

begin

a:= 256;

y:=512;

n:=1;

while a>0.99 do

begin

y:= a+ ((exp(ln(x)\*n)+a\*2)/y) ;

a:=a/2;

n:=n+1;

end;

y:= (exp(ln(x)\*n)+a\*2)/y ;

writeln('x= ',x:4:2, ' y= ',y:10:6);

x:=x+step;

end;

readln;

end.