# 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. 451001 |  | В. А. Пузик |
| Проверил |  | Асс. Е.Е. Фадеева |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Минск, 2018

1. Теоретические сведения по теме лабораторной работы

Типы циклов:

1. По типу вложенности
   1. Внешние циклы — циклы, не имеющие родительских, не вложенные в другой цикл
   2. Вложенный цикл — циклы, вложенные в другой цикл
   3. сложный цикл — конструкция, состоящая из внешнего и как минимум одного ложенного цикла
2. По типу счетчика
   1. цикл со счетчиком — цикл, выполняющийся фиксированное кол-во раз
   2. итерационный цикл — цикл, выполняющийся до тех пор, пока выполняется определенное условие, и количество итераций может быть неопределенным заранее.

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

Свойства правильного алгоритма:

1. Дискретность - любой алгоритм состоит из отдельных операций (этапов, действий), которые выполняются дискретно (по шагам).
2. Определенность — свойство алгоритма, которое гласит, что каждый его шаг должен быть строго определен и различные толкования должны быть исключены.
3. Результативность — свойство, которое гласит, что любой алгоритм должен завершаться с результатом.
4. Массовость — свойство, которое гласит, что любой алгоритм можно успешно применять к различным наборам исходных данных в определенном формате.
5. Задание на лабораторную работу
   1. Постановка задачи

Для заданного преподавателем пункта приведенной ниже таблицы вычислить значение функции f(x, n) для n = 10 11;... 15 и значения х, изменяющегося от x = 0.6 до x = 1.1 с шагом Delta x = 0.25 Результат вывести на печать в виде:

n = Значение x = Значение f = Значение

* 1. Эскиз ожидаемого результата

n=10 x=0.6 f=1.697

n=11 x=0.6 f=1.699

n=12 x=0.6 f=1.7

n=13 x=0.6 f=1.701

n=14 x=0.6 f=1.702

n=15 x=0.6 f=1.703

n=10 x=0.85 f=1.167

n=11 x=0.85 f=1.17

n=12 x=0.85 f=1.172

n=13 x=0.85 f=1.173

n=14 x=0.85 f=1.174

n=15 x=0.85 f=1.175

n=10 x=1.1 f=0.9071

n=11 x=1.1 f=0.9102

n=12 x=1.1 f=0.9126

n=13 x=1.1 f=0.9144

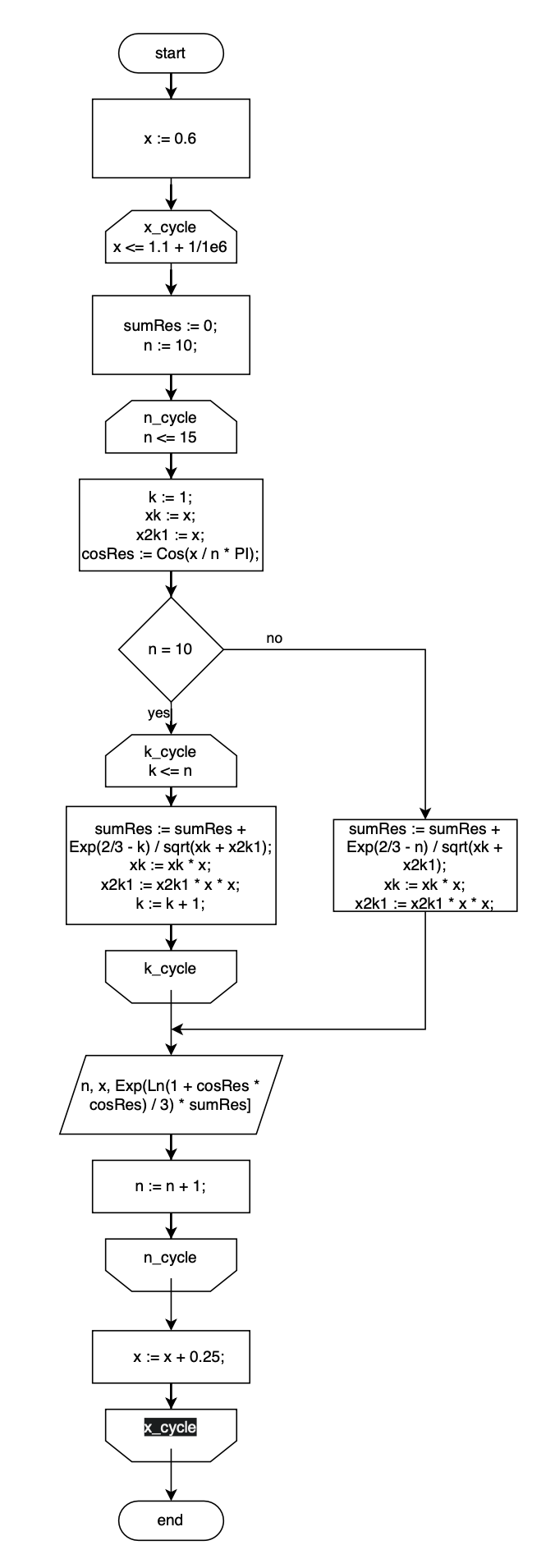
n=14 x=1.1 f=0.9159

n=15 x=1.1 f=0.9171

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 |
| n | Параметр дроби | int | 10 | 1) n := 10;  2) n := n + 1; | 1) x\_cycle  2) n\_cycle |
| x | Параметр дроби | double | 0.6 | x := x + 0.25; | X\_cycle |
| k | Параметр суммы | int | 1 | 1) k := 1;  2) k := k + 1; | 1) n\_cycle  2) k\_cycle |
| xk | Текушая степень x^k | double | x | xk := x  xk := xk \* x; | 1) n\_cycle  2) k\_cycle |
| x2k1 | Текушая степень x^(2k-1) | double | x | x2k1 := x;  x2k1 := x2k1 \* x \* x; | 1) n\_cycle  2) k\_cycle |
| sumRes | Текущая сумма | double | 0 | sumRes := 0;  sumRes := sumRes + Exp(2/3 - k) / sqrt(xk + x2k1);  sumRes := sumRes + Exp(2/3 - n) / sqrt(xk + x2k1); | 1) x\_cycle  2) k\_cycle  3) n\_cycle |
| cosRes | Результат вычисления части с косинусом | Double | - | cosRes := Cos(x / n \* PI); | N\_cycle |



* 1. Текст программы и его описание

Uses sysutils;

var

x: double;

n: Integer;

k: Integer;

xk: Double;

x2k1: Double;

sumRes: Double;

cosRes: Double;

begin

x := 0.6;

while x <= 1.1 + 1/1e6 do

begin

sumRes := 0;

n := 10;

while n <= 15 do

begin

k := 1;

xk := x;

x2k1 := x;

cosRes := Cos(x / n \* PI);

if n = 10 then

begin

while k <= n do

begin

sumRes := sumRes + Exp(2/3 - k) / sqrt(xk + x2k1);

xk := xk \* x;

x2k1 := x2k1 \* x \* x;

k := k + 1;

end

end

else

begin

sumRes := sumRes + Exp(2/3 - n) / sqrt(xk + x2k1);

xk := xk \* x;

x2k1 := x2k1 \* x \* x;

end;

WriteLn(Format('n=%d x=%.4g f=%.4g', [n, x, Exp(Ln(1 + cosRes \* cosRes) / 3) \* sumRes]));

n := n + 1;

end;

x := x + 0.25;

end;

end.

* 1. Тестирование и отладка программы

Таблица 3.2 Прохождение тестов программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Специфика тестирования | Номер теста | Вводимые данные | Ожидаемый результат | Полученный результат |
| Корректность работы программы | 1 | - | n=10 x=0.6 f=1.697  n=11 x=0.6 f=1.699  n=12 x=0.6 f=1.7  n=13 x=0.6 f=1.701  n=14 x=0.6 f=1.702  n=15 x=0.6 f=1.703  n=10 x=0.85 f=1.167  n=11 x=0.85 f=1.17  n=12 x=0.85 f=1.172  n=13 x=0.85 f=1.173  n=14 x=0.85 f=1.174  n=15 x=0.85 f=1.175  n=10 x=1.1 f=0.9071  n=11 x=1.1 f=0.9102  n=12 x=1.1 f=0.9126  n=13 x=1.1 f=0.9144  n=14 x=1.1 f=0.9159  n=15 x=1.1 f=0.9171 | n=10 x=0.6 f=1.697  n=11 x=0.6 f=1.699  n=12 x=0.6 f=1.7  n=13 x=0.6 f=1.701  n=14 x=0.6 f=1.702  n=15 x=0.6 f=1.703  n=10 x=0.85 f=1.167  n=11 x=0.85 f=1.17  n=12 x=0.85 f=1.172  n=13 x=0.85 f=1.173  n=14 x=0.85 f=1.174  n=15 x=0.85 f=1.175  n=10 x=1.1 f=0.9071  n=11 x=1.1 f=0.9102  n=12 x=1.1 f=0.9126  n=13 x=1.1 f=0.9144  n=14 x=1.1 f=0.9159  n=15 x=1.1 f=0.9171 |

* 1. Итоговый текст программы

Uses sysutils;

var

x: Double;

n: Integer;

k: Integer;

xk: Double;

x2k1: Double;

sumRes: Double;

cosRes: Double;

begin

x := 0.6;

while x <= 1.1 + 1/1e6 do

begin

sumRes := 0;

n := 10;

while n <= 15 do

begin

k := 1;

xk := x;

x2k1 := x;

cosRes := Cos(x / n \* PI);

if n = 10 then

begin

while k <= n do

begin

sumRes := sumRes + Exp(2/3 - k) / sqrt(xk + x2k1);

xk := xk \* x;

x2k1 := x2k1 \* x \* x;

k := k + 1;

end

end

else

begin

sumRes := sumRes + Exp(2/3 - n) / sqrt(xk + x2k1);

xk := xk \* x;

x2k1 := x2k1 \* x \* x;

end;

WriteLn(Format('n=%d x=%.4g f=%.4g', [n, x, Exp(Ln(1 + cosRes \* cosRes) / 3) \* sumRes]));

n := n + 1;

end;

x := x + 0.25;

end;

end.