Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Основы алгоритмизации и программирования (ОАиП)

ОТЧЕТ

по лабораторной работе №1

Тема работы: Расчет функции

Выполнил

студент: гр. 151004 Данилов Ф.А.

Проверил: Фадеева Е.П.

Минск 2021

Содержание

[1 Постановка задачи 3](#_Toc84109320)

[2 Текстовый алгоритм решения задачи 4](#_Toc84109321)

[3 Структура данных 6](#_Toc84109322)

[4 Схема алгоритма решения задачи по ГОСТ 19.701-90 7](#_Toc84109323)

[5 Результаты расчетов 9](#_Toc84109324)

[Приложение А 10](#_Toc84109325)

[Приложение Б 13](#_Toc84109326)

# Постановка задачи

Для заданной функции:

вычислить её значение для = 10;11…15 и значении , изменяющемся от = 0.6 до = 1.1 с шагом = 0.1.

Вывести на печать результаты расчётов:

= значение = значение = значение

# Текстовый алгоритм решения задачи

Таблица – Алгоритм решения

| Номер  шага | Назначение шага |
| --- | --- |
|  | xS := 0.6 |
|  | xF := 1.1 |
|  | xH := 0.1 |
|  | nS := 10 |
|  | nF := 15 |
|  | nH := 1 |
|  | x := xS |
|  | Начало цикла А1. Проверка выполнения условия (x <= xF). Если условие истинно, идти к шагу 9, иначе – к шагу 39 |
|  | atg := arctan(x) |
|  | flag := false |
|  | n := nS |
|  | Начало цикла А2. Проверка выполнения условия (n <= nF). Если условие истинно, идти к шагу 13, иначе – к шагу 37 |
|  | Проверка выполнения условия: ( n\*n\*n \* (x+1) <= 0). Если условие истинно, то идти к шагу 14, иначе – к шагу 17 |
|  | Вывод: 'Error. x = ', x, ', n = ', n |
|  | n := n + nH |
|  | Идти к шагу 12 |
|  | root := exp( 1/6 \* ln( (n\*n\*n) \* ( x + 1 ) ) ) |
|  | k := 1 |
|  | sum := 0 |
|  | Начало цикла А3. Проверка выполнения условия (k <= n). Если условие истинно, идти к шагу 21, иначе – к шагу 30 |
|  | Проверка выполнения условия: ( (cos( n \* x ) = 0) or (exp(-k\*x) + n \* exp((k-1)\*ln(x)) < 0) ). Если условие истинно, то идти к шагу 22, иначе – к шагу 25 |
|  | Вывод: 'Error. x = ', x, ', n = ', n |
|  | flag := true |
|  | Идти к шагу 30 |
|  | num := exp( 1 / 5 \* ln( exp(-k\*x) + n \* exp( (k-1)\*ln(x) ) ) ) |
|  | denum := cos( k \* x ) |
|  | sum := sum + num / denum |
|  | k := k + 1 |
|  | Идти к шагу 20 |
|  | Проверка выполнения условия: (flag). Если условие истинно, идти к шагу 31, иначе – к шагу 33 |
|  | n := n + nH |

Продолжение Таблицы 1

|  |  |
| --- | --- |
| Номер  шага | Назначение шага |
|  | Конец цикла А3. Идти к шагу 12 |
|  | f := atg \* root \* sum |
|  | Вывод: 'x = ', x:4:1, ', n = ', n, ', f = ', f:13:8 |
|  | n := n + nH |
|  | Конец цикла А2. Идти к шагу 12 |
|  | x := x + xH |
|  | Конец цикла А1. Идти к шагу 8 |
|  | Останов. |

# Структура данных

Таблица – Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| x | Real | Аргумент функции |
| xS | Real | *x*н |
| xF | Real | *x*к |
| xH | Real | *x* |
| n | Integer | Аргумент функции |
| nS | Integer | *n*н |
| nF | Integer | *n*к |
| nH | Integer | *n* |
| flag | Boolean | переменная для выхода из цикла |
| k | Integer | счетчик для вычисления суммы |
| atg | Real | arctan(x) |
| root | Real | (n\*(x+1)1/3)1/2 |
| num | Real | (e-k\*x +n\*xk-1)1/5 |
| denum | Real | cos(k\*x) |
| sum | Real | сумма дробей num/denum |
| f | Real | значение функции *f* |

# Схема алгоритма решения задачи по ГОСТ 19.701-90

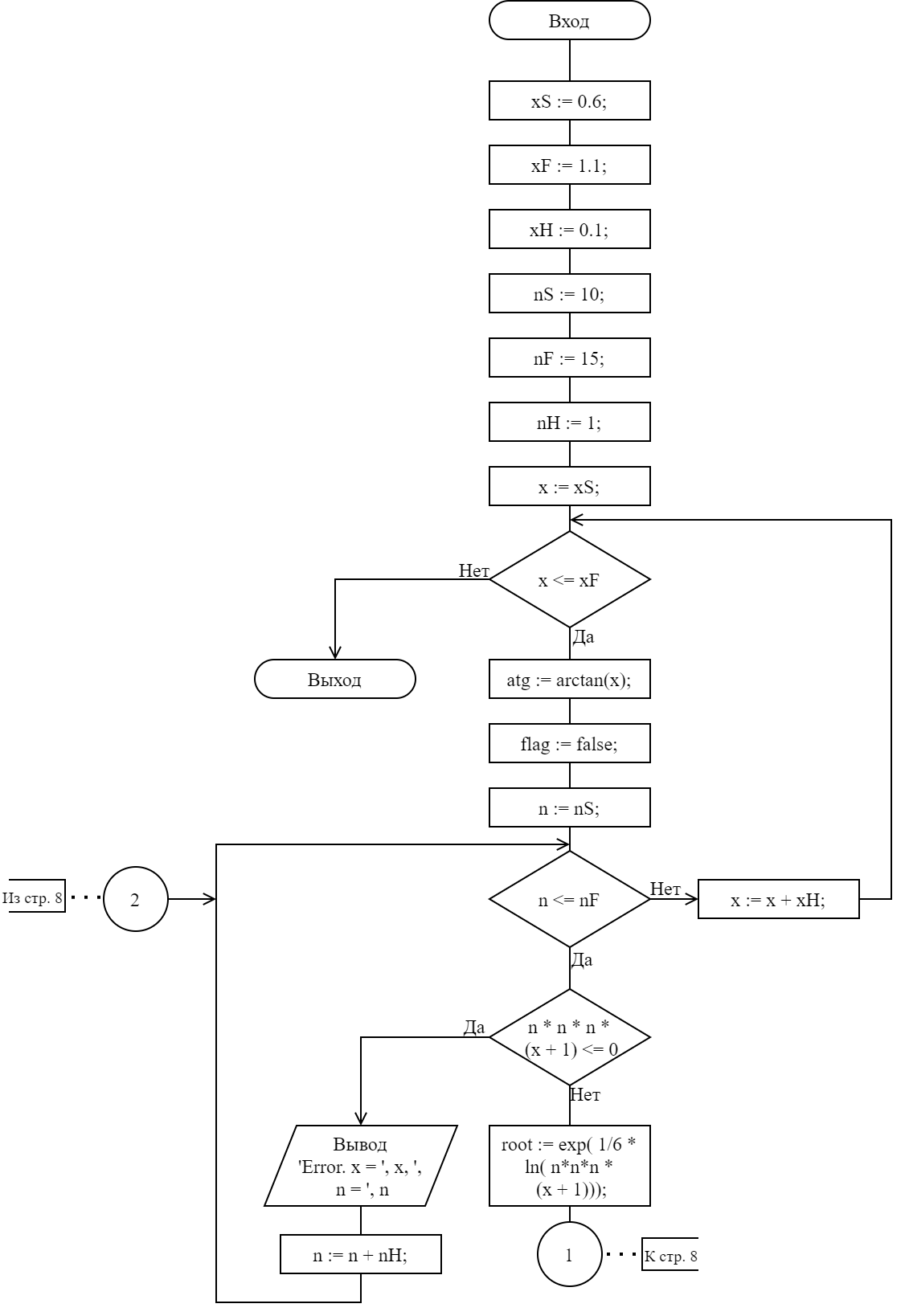


Рисунок 1 – Схема алгоритма решения задачи по ГОСТ 19.701-90 (часть 1)

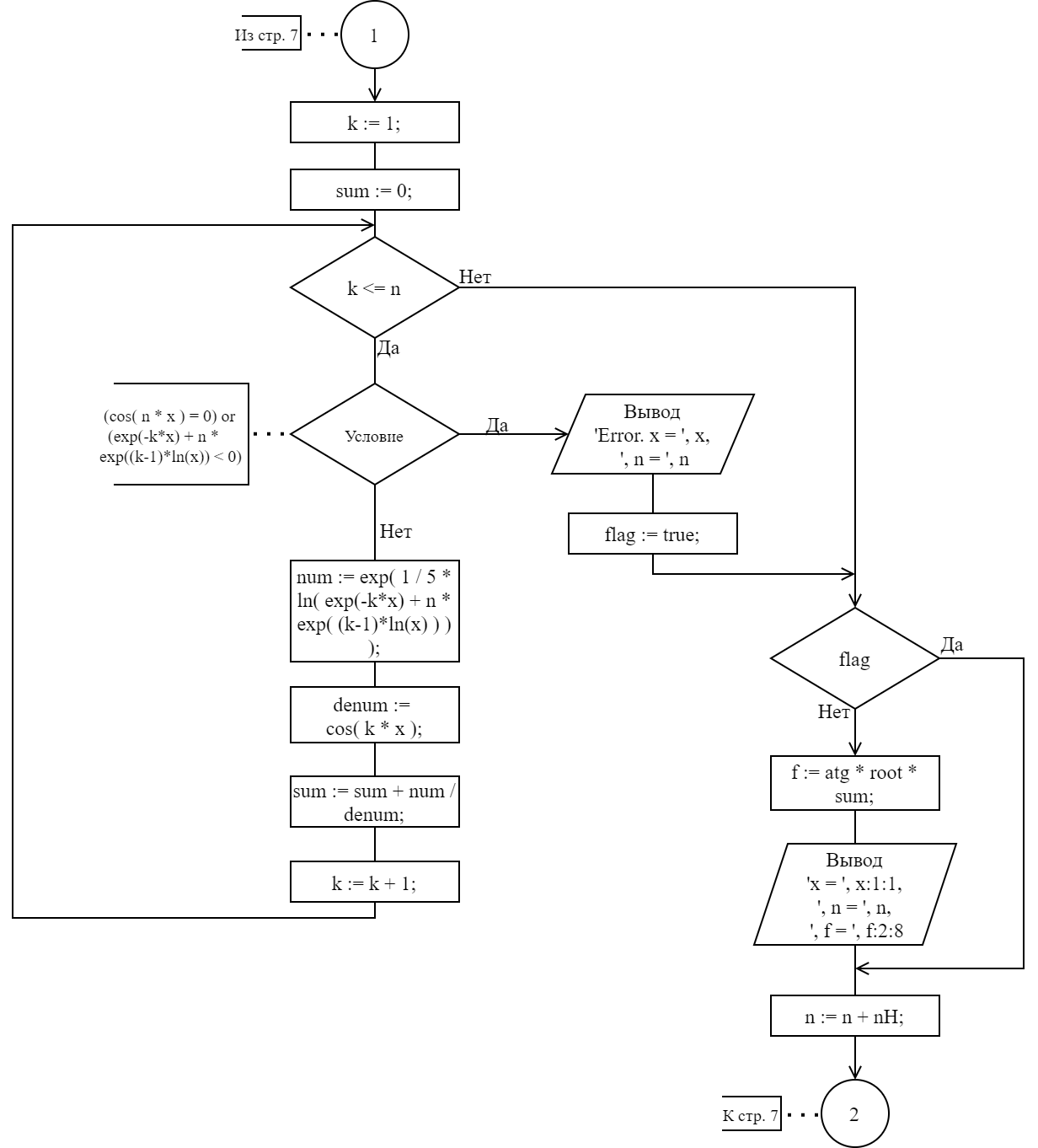


Рисунок 2 – Схема алгоритма решения задачи по ГОСТ 19.701-90 (часть 2)

# Результаты расчетов

Вследствие выполнения программы на экран выводятся следующие результаты:

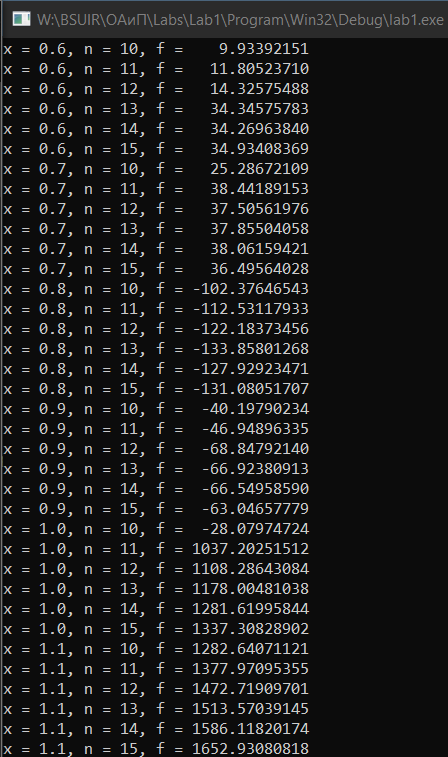


Рисунок 3 – Результаты расчётов

Приложение А

(обязательное)

Исходный код программы

program lab1;

uses

System.SysUtils;

var

xS, xF, x, xH: Real;

nS, nF, n, nH: Integer;

atg, root, sum, num, denum, f: Real;

flag: boolean;

k: Integer;

begin

// Initialize task variables

xS := 0.6;

xF := 1.1;

xH := 0.1;

nS := 10;

nF := 15;

nH := 1;

x := xS;

while (x <= xF) do

begin

// Calculate arctg(x)

atg := arctan(x);

// Reset Error boolean after loop

flag := false;

// Reset n to start value every loop

n := nS;

while (n <= nF) do

begin

// Exit loop if base of logarithm <= 0

if (n \* n \* n \* (x + 1) <= 0) then

begin

write('Error. x = ', x, ', n = ', n);

n := n + nH;

continue;

end;

// Calculate Power((n \* Power((x + 1), 1 / 3)), 1 / 2)

// using exp(power\*ln(base)) equivalent

root := exp( 1/6 \* ln( (n\*n\*n) \* ( x + 1 ) ) );

k := 1;

sum := 0;

while (k <= n) do

begin

// Exit this loop if denumerator is 0 or base of exp(power\*ln(base)) < 0

if ( (cos( n \* x ) = 0) or (exp(-k\*x) + n \* exp((k-1)\*ln(x)) < 0) ) then

begin

write('Error. x = ', x, ', n = ', n);

flag := true;

break;

end;

// Calculate the fraction

num := exp( 1 / 5 \* ln( exp(-k\*x) + n \* exp( (k-1)\*ln(x) ) ) );

denum := cos( k \* x );

// Add fraction to the sum

sum := sum + num / denum;

// Iterate k

k := k + 1;

end;

if (flag) then

begin

n := n + nH;

continue;

end;

// Calculate and output the result of multiplication

f := atg \* root \* sum;

writeln('x = ', x:1:1, ', n = ', n, ', f = ', f:2:8);

// Iterate n

n := n + nH;

end;

// Iterate x

x := x + xH;

end;

// Stop console from closing

Readln;

end.

Приложение Б

(обязательное)

Тестовые наборы

Таблица 3 – Тестовые наборы

| Тест | Исходные данные и ожидаемый  результат (Mathcad) | Полученный результат |
| --- | --- | --- |
| 1. 1. |  |  |
| 1. 2. |  |  |
| 1. 3. |  |  |

Продолжение таблицы 3.

|  |  |  |
| --- | --- | --- |
| Тест | Исходные данные и ожидаемый  результат (Mathcad) | Полученный результат |
| 1. 4. |  |  |
| 1. 5. |  |  |
| 1. 6. |  |  |