Содержание

[1 Постановка задачи 3](#_Toc121149431)

[1.1 Начальная постановка задачи 3](#_Toc121149432)

[1.2 Вывод результатa 3](#_Toc121149433)

[2 Методика решения 4](#_Toc121149434)

[2.1 Доказание сходимости бесконечного ряда 4](#_Toc121149435)

[2.2 Итерационные вычисления 4](#_Toc121149436)

[2.3 Проверка точности вычислений 5](#_Toc121149437)

[3 Текстовый алгоритм решения задачи 6](#_Toc121149438)

[4 Структура данных 8](#_Toc121149439)

[5 Схема алгоритма решения задачи по ГОСТ 19.701-90 9](#_Toc121149440)

[6 Результаты расчетов 11](#_Toc121149441)

[Приложение А 12](#_Toc121149442)

[Приложение Б 15](#_Toc121149443)

# Постановка задачи

## Начальная постановка задачи

Для заданной функции бесконечного ряда:

Рассчитать значение с точностями = 1\*10-5 и = = 1\*10-6 при X, изменяющемся от XStart = 0.1 до XFinish = 0.9 с шагом XStep = 0.1, используя цикл с постусловием Repeat…Until.

## Вывод результатa

Для двух заданных точностей вывести:

1. Значение переменной Х.
2. Значение функции бесконеного ряда Y.
3. Точность
4. Номер шага K, на котором это значение было получено.

# Методика решения

## Доказание сходимости бесконечного ряда

Сходимость бесконечного ряда была доказана с помощью признака Даламбера. Если существует предел отношения последующего члена к предыдущему , то:

Таблица – Условие сходимости по признаку Даламбера

|  |  |  |
| --- | --- | --- |
| № | Условие | Вывод |
|  | D < 1 | Ряд сходится |
|  | D > 1 | Ряд расходится |
|  | D = 1 | Признак Даламбера не даёт ответа. Нужно использовать другой |

Исследуем на сходимость ряд:

Найдём предел отношения последующего члена к предыдущему:

D < 1 для всех X в промежутке от 0.1..0.9 с шагом 0.1, следовательно, ряд сходится.

## Итерационные вычисления

Для расчёта значений бесконечного ряда используется итерационный цикл с постусловием Repeat…Until, условием выхода их которого будет разница между двумя последовательными значениями функции, меньшая или равная точности.

## Проверка точности вычислений

Проверка точности вычислений происходит путём сравнения разницы между двумя последовательными значениями функции элемента суммы с .

# Текстовый алгоритм решения задачи

Таблица – Алгоритм решения

|  |  |
| --- | --- |
| Номер  шага | Назначение шага |
|  | Acc1 = 1E-5 |
|  | Acc2 = 1E-6 |
|  | XStart = 0.1 |
|  | XFinish = 0.9 |
|  | XStep = 0.1 |
|  | X := XStart |
|  | I := 1 |
|  | Начало цикла А1. Проверка выполнения условия (X <= XFinish). Если условие истинно, то идти к шагу 9, иначе – к шагу 32 |
|  | Y0 := 0 |
|  | Y :=0 |
|  | K := 1 |
|  | Reach := false |
|  | Up := –X |
|  | Delta := Acc1 + 1 |
|  | Начало цикла А2 |
|  | Y0 := Y |
|  | Y := Y0 + Up / (K \* (K + 1) \* (K + 2)) |
|  | Up := Up \* X \* –X |
|  | Delta := Y – Y0 |
|  | Проверка выполнения условия (Not Reach and (Abs(Delta) <= Acc1)). Если условие истинно, то идти к шагу 21, иначе – к шагу 24 |
|  | Res1[i] := Y |
|  | Counter1[i] := k |
|  | Reach := True |
|  | K := K + 1 |
|  | Конец цикла А2. Проверка выполнения условия (Abs(Delta) <= Acc2). Если условие истинно, то идти к шагу 16, иначе – к шагу 26 |
|  | Res2[i] := Y |
|  | Counter2[i] := K |
|  | Reach:=True |
|  | X := X + XStep |
|  | I := I + 1 |
|  | Конец цикла А1. Идти к шагу 8 |
|  | Вывод Res1[1..9] |
|  | Вывод Counter1[1..9] |
|  | Вывод Res2[1..9] |
|  | Вывод Counter2[1..9] |
|  | Останов. |

# Структура данных

Таблица – Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| Res1 | Array [1 .. 9] of Real | Массив значений бесконечных рядов первой точности |
| Counter1 | Array [1 .. 9] of Integer | Массив количеств итераций бесконечных рядов первой точности |
| Res2 | Array [1 .. 9] of Real | Массив значений бесконечных рядов второй точности |
| Counter2 | Array [1 .. 9] of Integer | Массив количеств итераций бесконечных рядов второй точности |
| I | Integer | Счетчик цикла |
| K | Integer | Счётчик степеней бесконечного ряда |
| XStart | Real | Начальное значение Х |
| XFinish | Real | Конечное значение Х |
| XStep | Real | Шаг для изменения Х |
| Acc1 | Real | Первая точность для итерационных вычислений |
| Acc2 | Real | Вторая точность для итерационных вычислений |
| X | Real | Значение Х |
| Y0 | Real | Прошлое значание дроби |
| Y | Real | Значение дроби |
| Delta | Real | Разница между двумя последовательными членами бесконечного ряда |
| Up | Real | Значение числителя дроби |
| Reach | Boolean | Переменная для проверки достижения первой точности |

# Схема алгоритма решения задачи по ГОСТ 19.701-90



Рисунок – Схема алгоритма в соответствии с ГОСТ 19.701-90 (часть 1)



Рисунок – Схема алгоритма в соответствии с ГОСТ 19.701-90 (часть 2)

# Результаты расчетов

В результате выполнения программы выводятся следующие результаты расчётов:

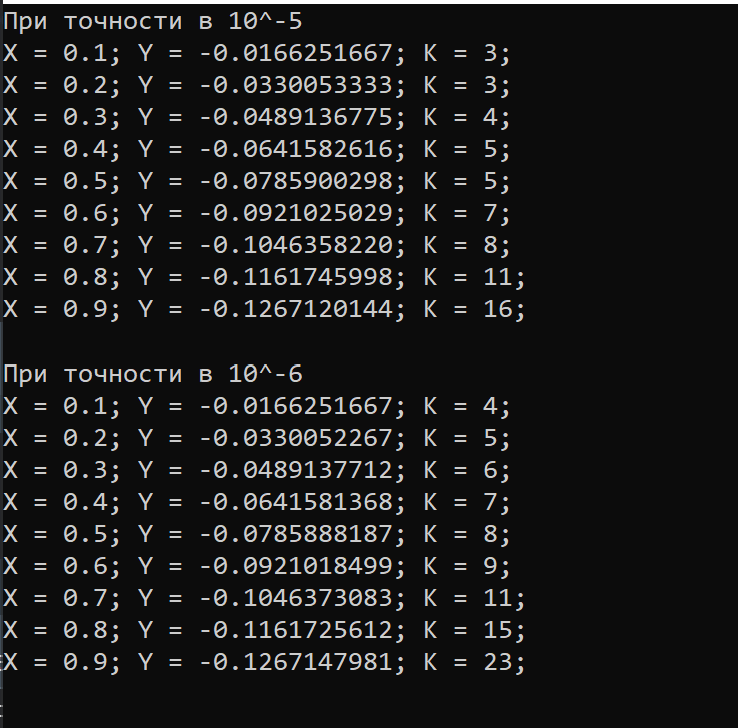


Рисунок 3 – Результаты расчётов

Приложение А

(обязательное)

Исходный код программы

Program Lab3;

{

For a given infinite function calculate its value for x ranging from 0.1 to 0.9 with a step of 0.1 and with

given precisions using the cycle "Repeat...Until"

}

// uses app

{$APPTYPE CONSOLE}

// declare consts

Сonst

Acc1 = 1E-5;

Acc2 = 1E-6;

XStart = 0.1;

XFinish = 0.9;

XStep = 0.1;

// XStart - start value of X

// XFinish - finish value of X

// XStep - step for incrementing X

// Acc1 - the first accuracy for calculating

// Acc2 - the second accuracy for calculating

Var

Res1: array [1 .. 9] of real;

Counter1: array [1 .. 9] of integer;

Res2: array [1 .. 9] of real;

Counter2: array [1 .. 9] of integer;

i, k: integer;

X, Sum, Prevsum, Up, Delta: real;

Reach: boolean;

// Res1 - First accuracy results

// Res2 - Second accuracy results

// Counter1 - first array of ks

// Counter2 - second array of ks

// X - variable X

// Y0 - previous function value

// Y - function value sum

// Up - numerator of the fraction

// Delta - delta for precised calculation

// K - counter for sum

// I - indicator for output

// Reach - needed accuracy reach tracker

Begin

// initialise i

I := 1;

// Start of Cycle A (x 0.1 to 0.9 in 0.1)

X := XStart;

While X <= XFinish do

begin

// reset y, y0, k

Y := 0;

Y0 := 0;

K := 1;

// initialise Delta

Delta := Acc1 + 1;

// initialize p

Up := -X;

//initialise reach

Reach := false;

// start of Cycle A1 (up to the specified accur)

repeat

// save the previous value of y

Y0 := Y;

// calculate the formula

Y := Y0 + Up / (K \* (K + 1) \* (K + 2));

// upgrade the numerator

Up := Up \* X \* (-X);

// calculate delta

Delta := Y – Y0;

// check for previous results and

// save the current

If not Reach and (abs(Delta) <= Acc1) then

begin

Res1[i] := Y;

Counter1[i] := K;

Reach := True;

end;

// increment K

K := K + 1;

until abs(Delta) <= Acc2;

// end of cycle A1

//save values of second type

Res2[i] := Y;

Counter2[i] := K;

Reach := True;

X := X + XStep;

I := I + 1;

// end of Cycle A

end;

writeln('При точности в 10^-5');

for I := 1 to 9 do

writeln('X = 0.', i, '; Y = ', Res1[i]:12:10,

'; K = ', Counter1[i], ';');

writeln;

writeln('При точности в 10^-6');

for i := 1 to 9 do

writeln('X = 0.', i, '; Y = ', Res2[i]:12:10, ';

K = ', Counter2[i], ';');

readln;

End.

Приложение Б

(обязательное)

Тестовые наборы

Таблица – Тестовые наборы

|  |  |  |
| --- | --- | --- |
| Тест | Данные по условию и ожидаемый  результат (Mathcad) | Полученный результат |
|  | X:= 0.1 | [См. пункт 6](#_Результаты_расчетов) |
|  | X:= 0.2 | [См. пункт 6](#_Результаты_расчетов) |
|  | X:=0.3 | [См. пункт 6](#_Результаты_расчетов) |
|  | X:=0.5 | [См. пункт 6](#_Результаты_расчетов) |
|  | X:=0.7 | [См. пункт 6](#_Результаты_расчетов) |