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 №3

Тема работы: Итерационные вычисления

Выполнил

студент: гр. 251003 Панкратьев Е.С.

Проверил: Фадеева Е.П.

Минск 2022

**СОДЕРЖАНИЕ**

[1 Постановка задачи 3](#_Toc119678999)

[2 Методика решения 4](#_Toc119679000)

[2.1 Доказательство сходимости ряда 4](#_Toc119679001)

[2.2 Краткое описание алгоритма решения 4](#_Toc119679002)

[3 Текстовый алгоритм решения задачи 5](#_Toc119679003)

[4 Структура данных 6](#_Toc119679004)

[5 Схема алгоритма решения задачи по ГОСТ 19.701-90 7](#_Toc119679005)

[6 Результаты расчетов 10](#_Toc119679006)

[Приложение А 11](#_Toc119679007)

[Приложение Б 14](#_Toc119679008)

# Постановка задачи

Вычислить значение функции бесконечного ряда:

Вычисления провести при значении , изменяющемся от до с шагом и точностях и .

Вывести на печать результаты расчётов по форме:

значение

значение значение k = значение

значение значение k = значение

# [Методика решения](#_Toc83996305)

## Доказательство сходимости ряда

Так как в числителе находится , ряд является знакопеременным. Значит, для доказательства сходимости ряда воспользуемся признаком сходимости Лейбница. Признак гласит: ряд сходится, а его сумма не превосходит первого члена, если одновременно выполняются следующие два условия:

1. Члены ряда монотонно убывают по абсолютной величине:  .
2. Предел его общего члена при неограниченном возрастании *n* равен нулю: .

**Доказательство:**

1. .
2. .

**Все условия соблюдаются при** .

Из этого следует, что данный ряд (при значении , изменяющемся от до с шагом ) является сходящимся.

## Краткое описание алгоритма решения

Для подсчета значении функции до точности Eps1 и Eps2, нужно достижения большей точности проверять в цикле (в данном примере Eps2), а достижение меньшей (в данном примере Eps1) проверять в теле цикла, используя логическую переменную.

Для расчета значений ряда воспользуемся циклом с предусловием while. Условие выполнения цикла – Delta > Eps2. Особенностью решения является необходимость перед входом в цикл переменной Delta присвоить значение Eps + 1. Это необходимо для входа в цикл.

Так как ряд является знакопеременным, стоит отметить, что при нечетном значении k элемент ряда будет отрицательным; при четном значении k – положительным. То есть, чтобы учитывать знакопеременность ряда, в теле цикла нужно числитель домножать на (-1).

Нахождение текущей точности происходит по определению: путем вычитания текущего значения суммы ряда из предыдущего (выражение необходимо взять в модуль из-за знакопеременности).

# Текстовый алгоритм решения задачи

Таблица 1 – Алгоритм решения

|  |  |
| --- | --- |
| Номер  шага | Назначение шага |
|  | Eps1 = 1E-5 |
|  | Eps2 = 1E-6 |
|  | xStart = 0.1 |
|  | xLast = 0.9 |
|  | xStep = 0.1 |
|  | x:= xStart |
|  | Начало цикла А1. Проверка выполнения условия (x <= xLast). Если условие истинно, перейти к шагу 8, иначе – к шагу 29 |
|  | Вывод х |
|  | y0:= 0 |
|  | k:= 0 |
|  | Num:= x\*x\*x |
|  | Den:= (4\*k+3)\*(4\*k+4) |
|  | FirstAccuracy:= False |
|  | Delta:= Eps2 + 1 |
|  | Начало цикла А1.1. Проверка выполнения условия (Delta > Eps2). Если условие истинно, перейти к шагу 16, иначе – к шагу 26 |
|  | y:= y0 + Num/Den |
|  | Delta:= abs(y - y0) |
|  | Если условия (Delta <= Eps1) и (FirstAccuracy = False) истинны, перейти к шагу 19. Иначе перейти к шагу 21 |
|  | Вывод Eps1, y, k |
|  | FirstAccuracy := True |
|  | y0:= y |
|  | k:= k + 1 |
|  | Num:= (-1) \* x \* Num |
|  | Den:= (4\*k+3)\*(4\*k+4) |
|  | Конец цикла А1.1. Вернуться к шагу 15 |
|  | Вывод Eps2, y, k |
|  | x:= x + xStep |
|  | Конец цикла А1. Вернуться к шагу 7 |
|  | Останов. |

# Структура данных

Таблица 2 – Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| Eps1 | Real | Первая точность до которой производится расчет |
| Eps2 | Real | Вторая точность до которой производится расчет |
| p | Integer | Количество всех символов для вывода |
| q | Integer | Количество символов после запятой для вывода |
| i | Integer | Счетчик цикла (текущее проверяемое число). |
| xStart | Real | Начальное значение аргумента функции |
| xLast | Real | Конечное значение аргумента функции |
| xStep | Real | Шаг для изменения х |
| x | Real | Значение аргумента функции |
| y | Real | Текущее значение функции |
| y0 | Real | Предыдущее значение функции |
| Num | Real | Значение числителя |
| Den | Real | Значение знаменателя |
| Delta | Real | Текущая точность |
| k | Integer | Счетчик для суммы |
| FirstAccuracy | Boolean | Индикатор нахождения первой точности |

# 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)



Рисунок 3 – Схема алгоритма решения задачи по ГОСТ 19.701-90 (часть 3)

# Результаты расчетов

После запуска программы на экран выводятся следующие результаты расчетов:

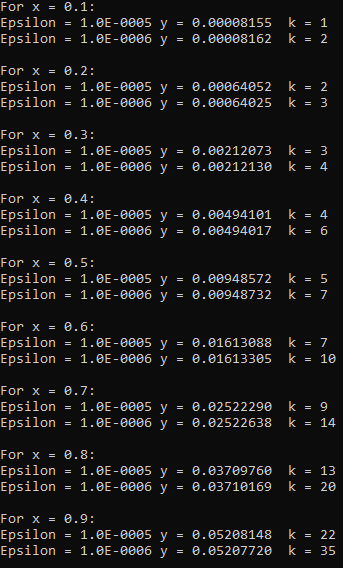


Рисунок 4 – Результаты расчетов

Приложение А

(обязательное)

Исходный код программы

Program Lab3Var1;

{

Calculate infinite function Y using precision Eps1 and Eps2.

}

{$APPTYPE CONSOLE}

Const

Eps1 = 1E-5;

Eps2 = 1E-6;

q = 8;

p = q + 2;

xStart = 0.1;

xLast = 0.9;

xStep = 0.1;

//Eps1, Eps2 - precision up to which the calculation is

//made

//q - an amount of simbols after comma

//p - an amount of simbols in a number

//xStart - start value for n,

//xLast - last value for x,

//xStep - step to change x.

Var

x, y, y0, Num, Den, Delta: real;

k: integer;

FirstAccuracy : boolean;

//x - function argument

//y - current function value

//y0 - previous function value

//Num - numerator value

//Den - denomerator value

//Delta - current precision

//k - counter for sum

//FirstAccuracy - flag to determine if the first accuracy is passed

Begin

//Initialize x

x:= xStart;

//Iterate over the value of x

while x<=xLast do

begin

//Display the current value of an argument

Writeln('For x = ',x:3:1,':');

//Since there is no previous function value, y0:= 0

y0:= 0;

//Initialize variables to count the first element

k:= 0;

Num:= x\*x\*x;

Den:=(4\*k+3)\*(4\*k+4);

//To start the cycle, reset FirstAccuracy

FirstAccuracy:= False;

//Since a cycle with a precondition is used,

//to enter the cycle, need to do the following:

Delta:= Eps2 + 1;

//Since Eps2 is less (need more precision), then in the

//cycle the condition will be it

while Delta > Eps2 do

begin

//Сount the current value of the function

y:= y0 + Num/Den;

//Count current precision

Delta:= abs(y - y0);

//Check if the first precision is reached

if (Delta <= Eps1) and (FirstAccuracy = False) then

begin

//Displaying the first value of the function with the

//first precision

Writeln('Epsilon =',Eps1:p,' y = ',y:p:q,' k = ',k);

//Mark that the value with the first precision is

//already displayed

FirstAccuracy := True;

end;

//Modernze variables for the next iteration

y0:= y;

k:= k + 1;

Num:= (-1) \* x \* Num;

Den:= (4\*k+3)\*(4\*k+4);

end;

//Displaying the second value of the function with the

//second precision

Writeln('Epsilon =',Eps2:p,' y = ',y:p:q,' k = ',k-1);

Writeln;

//Modernize x

x:= x + xStep;

end;

Readln;

End.

Приложение Б

(обязательное)

Тестовые наборы

Результаты расчетов

Таблица 3 - Результаты расчётов

|  |  |  |
| --- | --- | --- |
| Тест | Исходные данные и ожидаемый  результат (Mathcad) | Полученный результат |
| 1. |  | См. [главу 6](#_Результаты_расчетов) |
| 2. |  | См. [главу 6](#_Результаты_расчетов) |
| 3. |  | См. [главу 6](#_Результаты_расчетов) |

Продолжение таблицы 3

|  |  |  |
| --- | --- | --- |
| 4. |  | См. [главу 6](#_Результаты_расчетов) |
| 5. |  | См. [главу 6](#_Результаты_расчетов) |
| 6. |  | См. [главу 6](#_Результаты_расчетов) |

Продолжение таблицы 3

|  |  |  |
| --- | --- | --- |
| 7. |  | См. [главу 6](#_Результаты_расчетов) |
| 8. |  | См. [главу 6](#_Результаты_расчетов) |
| 9. |  | См. [главу 6](#_Результаты_расчетов) |