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 № 6

Тема работы: Работа с текстовыми файлами

Выполнил

студент: гр.551004 Довыдёнок М.А.

Проверил: Фадеева Е.П.

Минск 2016

Содержание

[1 Постановка задачи 3](#_Toc449519762)

[2 Описание алгоритмов 4](#_Toc449519763)

[3 Основные расчетные формулы 5](#_Toc449519764)

[4 Структура данных 6](#_Toc449519765)

[4.1 Структура данных основной программы 6](#_Toc449519766)

[4.2 Структура данных подпрограмм 6](#_Toc449519767)

[5 Схема алгоритма решения задачи по ГОСТ 19.701-90 8](#_Toc449519768)

[5.1 Схема основного алгоритма 8](#_Toc449519769)

[5.2 Схема алгоритма taylorSin(result, x, taylorEps) 9](#_Toc449519770)

[5.3 Схема алгоритма getResult(result, low, high, step, taylorEps) 10](#_Toc449519771)

[5.4 Схема алгоритма saveToFile(f, path, res) 11](#_Toc449519772)

[6 Результаты расчетов и тестирование программы 12](#_Toc449519773)

[Приложение А 13](#_Toc449519774)

# Постановка задачи

Создать текстовый файл F, напечатать в него таблицу значений функции sin(x) на отрезке [0..3] с шагом 0.1. Для расчёта функции sin(x) использовать её значения из библиотеки и написать расчёт, используя разложение функции sin(x) в ряд Тейлора с точностью eps = 0.0001.

Значения x печатать с одной цифрой в дробной части, значения синуса из библиотеки - с пятью, а значения синуса, рассчитанного через

ряд Тейлора – в экспоненциальной форме с 6 цифровыми знаками в мантиссе.

# Описание алгоритмов

Таблица – Описание алгоритмов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование  алгоритма | Назначение  алгоритма | Формальные  параметры | Рекомендуемый тип |
|  | Основной алгоритм | Вызывает getResult и saveToFile |  |  |
|  | taylorSin  (result, x, taylorEps) | Вычисляет sin(x) с помощью разложения в ряд Тейлора | result, x, taylorEps.  Возвращаемый параметр: result | Функция.  Возвращаемый параметр: result |
|  | getResult (result, low, high, step, taylorEps) | Заполняет массив result, вызывает библиотечный алгоритм sin и taylorSin для x изменяющегося от low до high с шагом step | result, low, high, step, taylorEps.  Возвращаемый параметр: result | Функция.  Возвращаемый параметр: result |
|  | saveToFile(f, path, res) | Возвращает матрицу result, равную произведению числа num и матрицы a | f, path, res | Процедура |

# Основные расчетные формулы

## Разложение функции sin в ряд Маклорена:

# Структура данных

## Структура данных основной программы

Таблица 2 – Структура данных основной программы

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| LOW | Real = 0 | Нижняя граница вычисляемой области |
| HIGH | Real = 3 | Верхняя граница вычисляемой области |
| STEP | Real = 0.1 | Шаг изменения значения в вычисляемой области |
| TAYLOR\_EPS | Real = 0.0001 | Точность расчёта sin с использованием разложения в ряд Тейлора |
| FILE\_PATH | String = “sin0\_3.txt” | Путь к файлу |
| f | Text | Текстовый файл |
| res | Array [0..30, 1..3] of Real | Массив, содержащий результаты расчётов |

## Структура данных подпрограмм

Таблица 3 – Структура данных алгоритма taylorSin(result, x, taylorEps)

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| result | Real | Значение sin(x) |
| x | Real | Значение x |
| taylorEps | Real | Точность расчёта sin с использованием разложения в ряд Тейлора |
| elem | Real | Текущий элемент ряда |
| fact | Integer | Факториал |

Таблица 4 – Структура данных алгоритма

getResult(result, low, high, step, taylorEps)

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| result | Array [0..30, 1..3] of Real | Результаты вычислений |
| low | Real | Нижняя граница вычисляемой области |
| high | Real | Верхняя граница вычисляемой области |
| step | Real | Шаг изменения значения в вычисляемой области |
| taylorEps | Real | Точность расчёта sin с использованием разложения в ряд Тейлора |
| current | Real | Текущее значение X |
| i | Integer | Счетчик цикла |

Таблица 5 – Структура данных алгоритма saveToFile(f, path, res)

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| f | Text | Текстовый файл |
| path | String | Путь к файлу |
| res | Array [0..30, 1..3] of Real | Результаты расчётов |
| i | Integer | Счетчик цикла |

# Схема алгоритма решения задачи по ГОСТ 19.701-90

## Схема основного алгоритма



Рисунок 1 – Схема основного алгоритма

## Схема алгоритма taylorSin(result, x, taylorEps)



Рисунок 2 – Схема алгоритма taylorSin(result, x, taylorEps)

## Схема алгоритма getResult(result, low, high, step, taylorEps)



Рисунок 3 – Схема алгоритма getResult(result, low, high, step, taylorEps)

## Схема алгоритма saveToFile(f, path, res)



Рисунок 4 – Схема алгоритма saveToFile(f, path, res)

# Результаты расчетов и тестирование программы

Ожидаемый результат:

Создан файл sin0\_3.txt, значения во втором и третьем

столбцах файла различаются не более чем на 0.0001

Полученный результат:

Рисунок 5 – Файл sin0\_3.txt

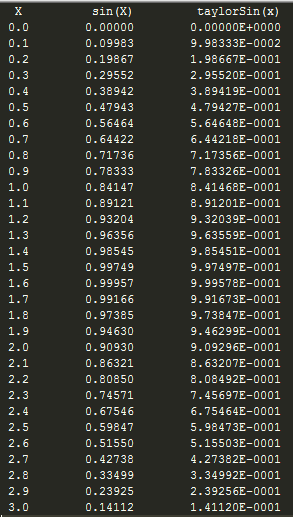


Рисунок 5 – Файл sin0\_3.txt

Приложение А

(обязательное)

Исходный код программы

program Lab12;

{$APPTYPE CONSOLE}

uses

SysUtils;

const

// Диапазон

LOW = 0;

HIGH = 3;

// Шаг вычислений

STEP = 0.1;

// Точность вычисления

TAYLOR\_EPS = 0.0001;

// Путь к файлу

FILE\_PATH = 'sin0\_3.txt';

type

TResult = array [0..30, 1..3] of Real;

var

f: Text;

res: TResult;

// Вычисляет значение sin с помощью разложения в ряд Тейлора

function taylorSin(x, taylorEps: Real):Real;

var

elem: Real;

fract: Integer;

begin

result := 0;

fract := 1;

elem := x;

repeat

result := result + elem;

elem := - elem \* x \* x;

elem := elem / ((fract + 1) \* (fract + 2));

inc(fract, 2);

until abs(elem) < taylorEps / 10;

end;

// Заполняет массив результатами

function getResult(low, high, step,

taylorEps: Real):TResult;

var

current: Real;

i: Integer;

begin

current := low;

for i := 0 to 30 do

begin

result[i, 1] := current;

result[i, 2] := sin(current);

result[i, 3] := taylorSin(current, taylorEps);

current := current + step;

end;

end;

// Создаёт файл и записывает туда результаты

procedure saveToFile(var f: text; path: String;

res: TResult);

var

i: Integer;

begin

AssignFile(f, path);

Rewrite(f);

DecimalSeparator := '.';

writeln(f, 'X':3, 'sin(X)':16, 'taylorSin(x)':21);

for i := 0 to 30 do

writeln(f, res[i, 1]:4:1,

res[i, 2]:15:5,

floatToStrF(res[i, 3], ffExponent, 6, 4):21);

CloseFile(f);

end;

begin

res := getResult(LOW, HIGH, STEP, TAYLOR\_EPS);

saveToFile(f, FILE\_PATH, res);

end.