**Лабораторная работа 5**

**Вычисление функции разложением ее в ряд**

**Для заданного одномерного массива cоставить алгоритм и программу нахождения суммы ряда с заданной точностью . Использовать рекуррентные соотношения при вычислении очередного элемента ряда. Для приведенных ниже рядов сходимость доказана при .**

|  |  |  |
| --- | --- | --- |
| 2 |  |  |

* Для сумма S = 0.4609119862
* Вывод формулы:
* Проверка для

|  |  |
| --- | --- |
| k = 1 |  |
| k = 2 |  |
| k = 3 |  |

* E = 10^-2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| №  Номер  элемента  массива X | X | S(x),  сумма ряда | K,  число потребовавшихся итераций | F(x),  контрольная формула | , контроль точности |
| 1 | -0.98 | 1.461 | 3 | 1.431 | 0.02976 |
| 2 | -0.5 | 0.461 | 3 | 0.461 | 0.00016 |
| 3 | 0.1 | 0.020 | 2 | 0.020 | 0.00000 |
| 4 | 0.5 | 0.461 | 3 | 0.461 | 0.00016 |
| 5 | 0.95 | 1.391 | 3 | 1.367 | 0.02345 |

* E = 10^-4

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| №  Номер  элемента  массива X | X | S(x),  сумма ряда | K,  число потребовавшихся итераций | F(x),  контрольная формула | , контроль точности |
| 1 | -0.98 | 1.43210 | 5 | 1.43115 | 0.0009566 |
| 2 | -0.5 | 0.46090 | 4 | 0.46091 | 0.0000078 |
| 3 | 0.1 | 0.01993 | 2 | 0.01993 | 0.0000002 |
| 4 | 0.5 | 0.46090 | 4 | 0.46091 | 0.0000078 |
| 5 | 0.95 | 1.36785 | 5 | 1.36719 | 0.0006637 |

* E = 10^-6

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| №  Номер  элемента  массива X | X | S(x),  сумма ряда | K,  число потребовавшихся итераций | F(x),  контрольная формула | , контроль точности |
| 1 | -0.98 | 1.4310143 | 6 | 1.4311477 | 0.000133327 |
| 2 | -0.5 | 0.4609042 | 4 | 0.4609120 | 0.000007813 |
| 3 | 0.1 | 0.0199335 | 3 | 0.0199335 | 0.000000000 |
| 4 | 0.5 | 0.4609042 | 4 | 0.4609120 | 0.000007813 |
| 5 | 0.95 | 1.3671034 | 6 | 1.3671902 | 0.000086849 |

* E = 10^-8

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| №  Номер  элемента  массива X | X | S(x),  сумма ряда | K,  число потребовавшихся итераций | F(x),  контрольная формула | , контроль точности |
| 1 | -0.98 | 1.431163868 | 7 | 1.431147664 | 0.00001620484 |
| 2 | -0.5 | 0.460912314 | 5 | 0.460911986 | 0.00000032735 |
| 3 | 0.1 | 0.019933508 | 3 | 0.019933508 | 0.00000000042 |
| 4 | 0.5 | 0.460912314 | 5 | 0.460911986 | 0.00000032735 |
| 5 | 0.95 | 1.367200137 | 7 | 1.367190224 | 0.00000991248 |

//-----------------------------------------------------------------------

#include <stdio.h> // printf, scanf

#include <conio.h> // getch

#include <math.h> // fabs, pow, ceil\floor, log, log10, exp, sqrt

#include <stdlib.h> // randomize, rand

#include <windows.h> // SetConsoleOutputCP(1251); SetConsoleCP(1251);

#include <time.h> //Time for random

//---------------------------------------------------------------------------

const double xx[7] = { 0.00001, -0.99, -1, -0.1, 0.1, 1, 0.99 };

#pragma warning(disable: 4996)

#pragma argsused

void main()

{

int n, i, k, z;

double e, sl, s2, sum, f, \*x; // четыре простого типа и один – дин.массив(указатель)

SetConsoleOutputCP(1251);

srand(time(NULL));

printf("Введите e=? ");

scanf("%lf", &e); // ввод числа с плавающей точкой двойной точности(%lf) e

if (e < 1e-13 || e>0.11) {

printf("Некорректная точность e (0..0.1] \nPress any key");

\_getch(); // ожидание нажатия клавиши

return; // выход из функции main

}

fflush(stdin); // очищаем буфер (лишние символы после ввода e)

printf("Введите n=? ");

scanf("%d", &n); // ввод десятичного(%d) n

if (n < 1 || n>20) {

printf("Invalid n [1..20]! \nPress any key");

\_getch(); // ожидание нажатия клавиши

return; // выход из функции main

}

fflush(stdin);

x = new double[n]; // выделяем память для n элементов массива

printf("Введите n=%d значений X из интервала (-1,+1):\n", n);

for (i = 0; i < n; i++) { // ввод x[0]...x[n-1]

scanf("%lf", &x[i]); // типа long float (%lf)

if (fabs(x[i]) >= 1) {

x[i] = xx[rand() % 7]; // одно из семи значений массива xx

if (fabs(x[i]) == 1) {

x[i] = x[i] \* (rand() % 100) / 100;

if (x[i] == 0) x[i] = e; // для некоторых вариантов x=/=0

}

printf("Некорректное значение заменено на %15.10lf\n", x[i]);

}

}

z = ceil(fabs(log(e) / log(10.0))) + 1; //Округление до ближайшего целого числа, не меньшего исходного

printf("e = %\*.\*lf\n", z + 2, z, e); // вывод e:(z+2):z

printf("N | X | Sum(X) | K| F(X) | |Sum(X)-F(X)|\n");

for (i = 1; i < 80; i++) printf("="); printf("\n");

for (i = 0; i < n; i++) {

// поиск суммы ряда

sl = s2 = pow(x[i], 2); // sl=pow(x[i],2); sum=sl; // первое слагаемое

sum = sl + s2;

k = 1;

while ((fabs(sl) >= e) && (k != 500)) {

sl \*= -pow(x[i], 2) / (4 \*pow(k, 2) + 2 \* k);

s2 \*= -pow(x[i], 2) / (k + 1);

sum += sl + s2;

k++;

};

f = x[i] \* sin(x[i]) - pow(exp(1.0), -pow(x[i], 2)) + 1;

printf("%2d|%17.\*lf|%17.\*lf|%2d|%17.\*lf|%17.\*lf\n",

i + 1, z, x[i], z, sum, k, z, f, z + 2, fabs(sum - f));

}

delete[] x; // освобождение памяти для одномерного дин.массива

printf("Нажмите любую клавишу - Press any key");

\_getch();

return;

}