

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

**Кафедра системного програмування та спеціалізованих комп’ютерних систем**

**Лабораторна робота №1**

з дисципліни

**«Алгоритми і методи обчислень »**

**Тема «Обчислення значень функції»**

Виконала: студентка ІII курсу ФПМ

групи КВ-72

Дорош К.А.

Перевірив:

**Київ 2019**

**Завдання для лабораторної роботи**

Для заданого варіанту (табл. 1.1) виконати 3 завдання.

1. Побудувати таблицю залежності довжини ряду n, що забезпечує точність не меншу за задане значення (*eps*) у точці *x = (b + a) / 2*, від *eps*:

|  |  |  |  |
| --- | --- | --- | --- |
| *eps* | n | Величина абсолютної похибки | Величина залишкового члена |
| 1.e-2 | 4 | 0.005 | 0.001 |
| ... | ... | ... | ... |
|  |  |  |  |

Значення *eps* змінюється від 1e-2 до 1e-14 з кроком 1e-3.

2. Для n (тобто довжина ряду фіксована й дорівнює n), отриманого в п.1. при *eps* = 1e-8, у точках *xi = a + h\*i, h = (b - a)/10, i = 0,...10* обчислити абсолютну похибку і залишковий член ряду. Результати подати у вигляді таблиці:

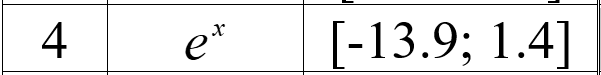
|  |  |  |
| --- | --- | --- |
| *xi* | Абс. похибка | Зал. Член |
| 0 | 0.005 | 0.001 |
| ... | ... | ... |
|  |  |  |

П р и м і т к а.. Точним значенням функції вважати результат, що його дає бібліотечна функція Сі, при обчисленні якої утримуються всі члени ряду, відмінні від нуля.

3. За допомогою AdvancedGrapher побудувати графік залежності абсолютної похибки від *x* (у логарифмічному масштабі).

Звіт має містити вихідний текст програми, таблицю з результатами та висновки.

**Варіант 4**



**// header.h**

#include <iostream>

#include <cmath>

#include <iomanip>

#define a -13.9

#define b 1.4

#define table\_len 4

using namespace std;

double Rn\_calc(double eps, int& n, double& x, double& result);

**// calculation.cpp**

#include "header.h"

double Rn\_calc(double eps, int& n, double& x, double& result) {

double Un = 1, int\_x = 1, float\_x = 1, Rn = 1, divx = 0, modx = 0;

divx = modf(x, &modx); // [x] + {x}

int\_x = exp(double(modx));// e^[x]

while (abs(Un) >= eps) { //while Un >= epsilon

Un = divx \* Un / n; // Uk = [x] \* Uk-1 / k

float\_x = float\_x + Un; // Sk = Sk-1 + Uk

n++; //k++

}

//Sk == e^{x}

result = float\_x \* int\_x; //e^x = e^[x]\*e^{x}

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

Rn = Rn \* abs(divx) / (i + 1); // R = |x|^n+1 /(n+1)!

}

Rn = Rn \* (n + 1) / n; // R = R \* n+1 / n

return Rn;

}

**//main.cpp**

#include "header.h"

int main() {

double eps = 0.01, Rn = 0, abserr = 0, res = 0;

int n = 1;

double x = (a + b) / 2;;

double h = (b - a) / 10;

cout << "|------------------------------------------------------------------------\n";

cout << setw(40) << "TABLE1" << setw(33) << "|" << endl;

cout << "|-----------------------------------------------------------------------|\n";

cout << "| Epsilon | n | Absolute Error | Rn | Result |\n";

for (int i = 0; i <= table\_len; i++) {

Rn = Rn\_calc(eps, n, x, res);

abserr = abs(res - exp(x));

cout << "|-----------------------------------------------------------------------|\n";

printf("|%13e | %3d | %15e | %15e | %13f|\n", eps, n, abserr, Rn, res);

eps = eps / 1000;

n = 1;

res = 0;

}

cout << "-------------------------------------------------------------------------\n";

cout << "\n\n ---------------------------------------------------------\n";

cout << "|" << setw(45) << "TABLE2 (eps = 1.0e-008)" << setw(12) << "|" << endl;

cout << "|--------------------------------------------------------|\n";

cout << "| x | Absolute Error | Rn | Result |" << endl;

eps = 0.00000001;

for (int i = 0; i <= 10; i++) {

x = a + h \* i;

Rn = Rn\_calc(eps, n, x, res);

abserr = abs(res - exp(x));

cout << "|--------------------------------------------------------|\n";

printf("|%6.2f %15e %16e %16f|\n", x, abserr, Rn, res);

n = 1;

res = 0;

}

cout << "---------------------------------------------------------\n";

return 0;

}

**Результат роботи програми**

