Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

**Алгоритмізація та програмування 1: Базові концепції програмування**

ЗВІТ

до лабораторної роботи №5

«Рекурсивний виклик функції»

*(ТЕМА)*

Варіант № 1

Дата «24» листопада 2021 Виконав: студент 1 курсу

гр. ТР-15

Руденко Владислав Ігорович

*(П.І.Б.)*

Оцінка «\_\_\_\_\_\_\_\_\_» Перевірив:

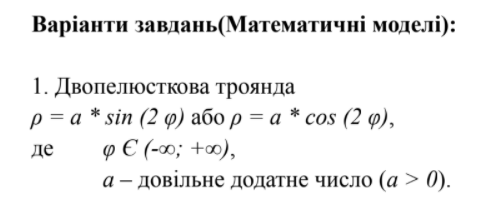
Крячок Олександр Степанович

Дата «\_\_\_» \_\_\_\_\_\_\_ 20\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(П.І.Б., підпис)*

Київ – 2021

***Завдання до роботи:***  
1. Ознайомитись з рекурсивним викликом функції.  
2. Розробити алгоритмом розрахунку значення функції за її розкладенням у ряд за умови отримання результату з заданою  точністю. Врахувати діапазон дозволених значень для змінної *x*.  
3. У якості індивідуального завдання необхідно написати програмний код, що реалізує алгоритм розрахунку значень функцій за їх розкладенням в ряд із заданою користувачем точністю.



***Теоритичні відомості***

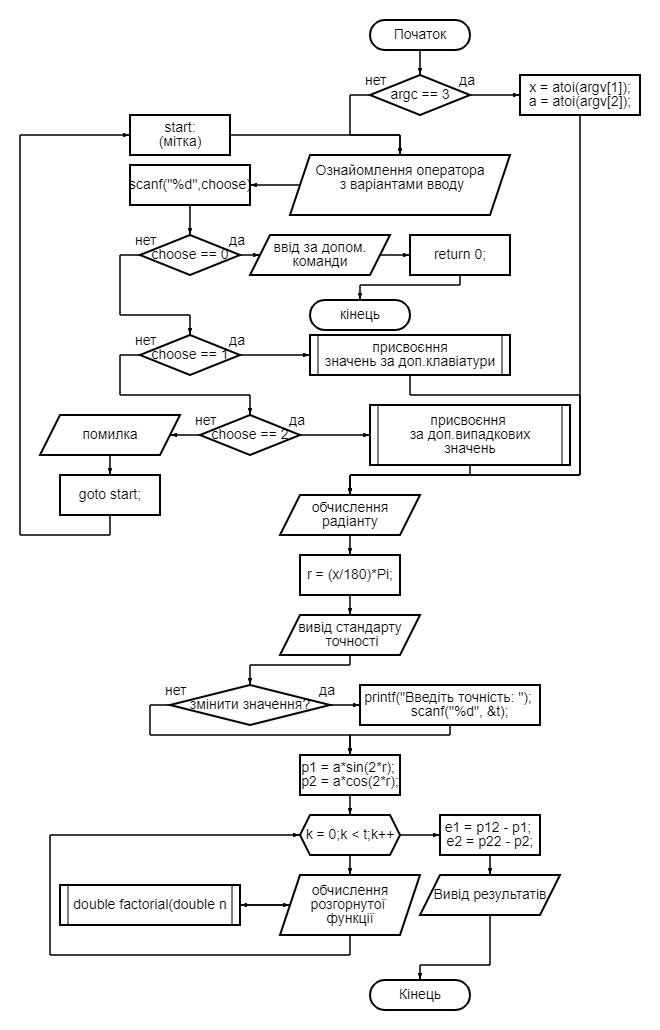
Рекурсивні функції — клас функцій, введений як уточнення класу обчислюваних функцій. В математиці загальноприйнятою є теза про те, що клас функцій, для обчислення яких існують алгоритми, при найширшому розумінні алгоритму, збігається з класом рекурсивних функцій. У зв'язку з цим, рекурсивні функції грають важливу роль в математиці та її застосуваннях, в першу чергу, в математичній логіці, основах математики та кібернетиці, як ефективно обчислювані функції.

Рекурсивні функції в програмуванні

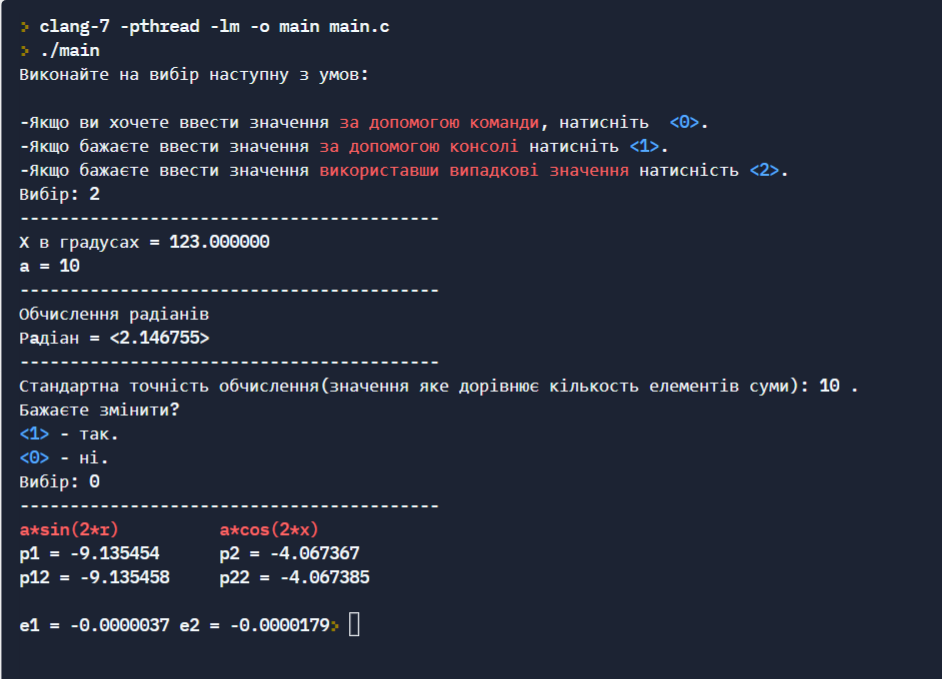
У програмуванні рекурсія — виклик функції чи процедури з неї самої (звичайно з іншими значеннями вхідних параметрів), чи безпосередньо через інші функції (наприклад, функція А викликає функцію B, а функція B — функцію A). Кількість вкладених викликів функції чи процедури називається глибиною рекурсії.  
Міць рекурсивного визначення об'єкта в тім, що таке кінцеве визначення здатне описувати нескінченно велике число об'єктів. За допомогою рекурсивної програми ж можливо описати нескінченне обчислення, причому без явних повторень частин програми.  
Варто уникати надлишкової глибини рекурсії, тому що це може викликати переповнення стека викликів.

Факторіал - функція, визначена на безлічі невід'ємних цілих чисел. Назва походить від латів. factorialis - діючий, що виробляє, множить; позначається n!, вимовляється ен факторіал. Факторіал натурального числа n визначається як добуток усіх натуральних чисел від 1 до n.

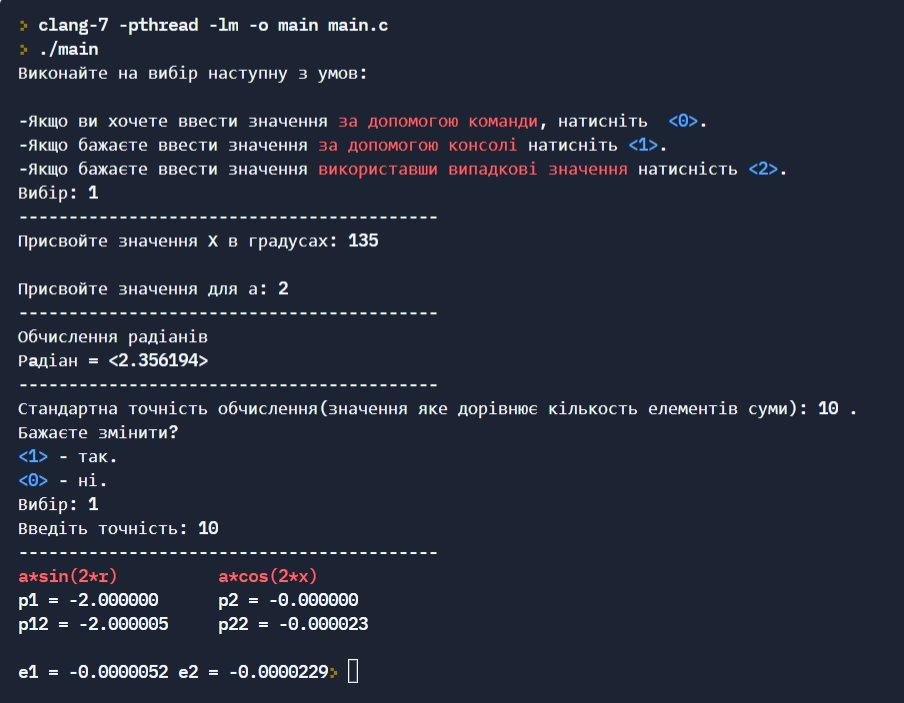
***Блок схема алгоритму***



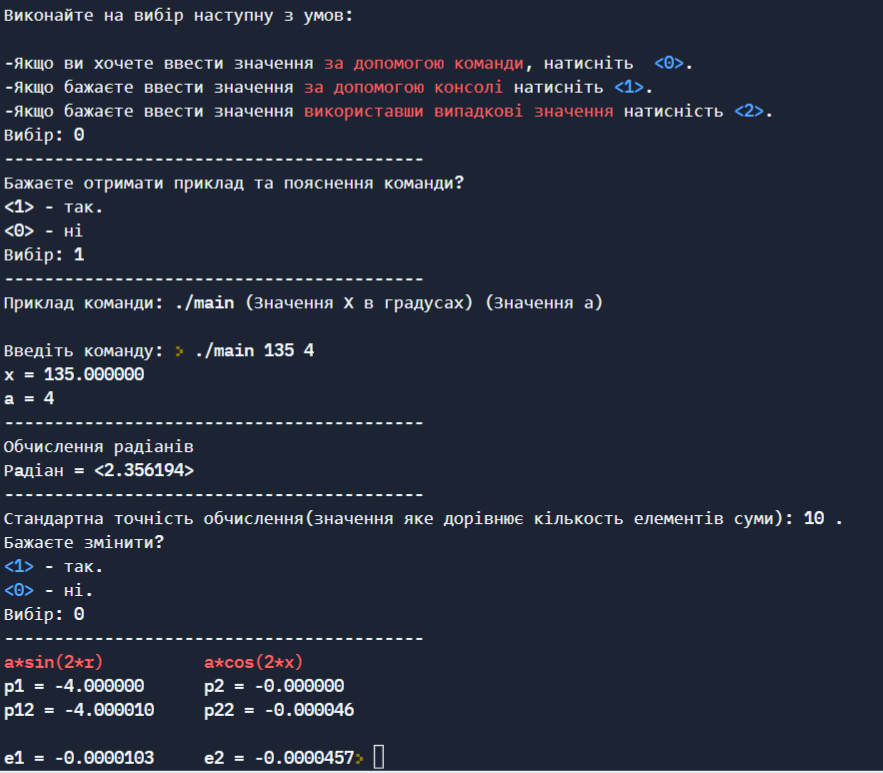
***Результати роботи:***

**

Результат завдання 1

**

Результат завдання 2

**

Результат завдання 3

*Хід виконаної роботи:*

Спочатку було створенно 3 варіанти вибору вводу, які надавались оператору, після вибору було виконано присвоєння значень,далі значення х обробляється та перетворюється в значення радіанту.

Повідомлення до оператора щодо точності, якщо оператора влаштовує програма продовжує хід, слідом йде обчислення базового завдання після - обчислення розкладеної функції за допомогою окремого методу для вираження факторіалу, в кінці програма виводить значення.

***Посилання на програму в repl.it:***

*<https://replit.com/join/hkwizmqezu-hetik>*

***Висновки:***

В ході виконання лабораторної роботи №5 було покращено навички користуванням математичних функцій в рамках мови програмування Сі, під час роботи було розглянуто розгорнуту функцію на базі двопелюсткової рози. Під час виконання було виявлено та виправлено n-на кількість помилок. За рахунок минулих ЛР було скорочено деякі аспекти коду, що робить його більш компактнішим.

***Програмний код (у текстовому варіанті):***

///24.11.21////Rudenko//Vladislav//LR5\_1///

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

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <math.h>

#define RED "\x1b[31m"

#define BLUE "\x1b[34m"

#define YELLOW "\033[1;33m"

#define RESET "\x1b[0m"

#define Pi 3.1415926

double factorial(double n);

int main(int argc, char \*argv[], char \*env[]) {

srand(time(0));

int choose, loky, a, i, t;

double r, x;

//////////////////////////////////////////////////////////////////////////////////

if(argc == 3){

x = atoi(argv[1]);

a = atoi(argv[2]);

if(a < 0){

printf("Error... a повинно бути більшим за 0.\n");

goto ch0;

}

printf("x = %f\na = %d\n------------------------------------------\n", x, a);

}

else{

start:

printf("Виконайте на вибір наступну з умов:\n\n-Якщо ви хочете ввести значення "RED"за допомогою команди"RESET", натисніть "BLUE"<0>"RESET".\n-Якщо бажаєте ввести значення "RED"за допомогою консолі"RESET" натисніть "BLUE"<1>"RESET".\n-Якщо бажаєте ввести значення "RED"використавши випадкові значення"RESET" натисність "BLUE"<2>"RESET".\nВибір: ");

scanf("%d", &choose);

printf("------------------------------------------\n");

if(choose == 0){

printf("Бажаєте отримати приклад та пояснення команди?\n<1> - так.\n<0> - ні\nВибір: ");

scanf("%d", &loky);

if(loky == 1) printf("------------------------------------------\nПриклад команди: ./main (Значення Х в градусах) (Значення а)\n\n");

ch0:

printf("Введіть команду: ");

return 0;

}

else if (choose == 1) {

printf("Присвойте значення Х в градусах: ");

scanf("%lf", &x);

ch1:

printf("\nПрисвойте значення для а: ");

scanf("%d", &a);

if(a < 0){

printf("Error... a повинно бути більшим за 0.\n");

goto ch1;

}

printf("------------------------------------------\n");

}

else if (choose == 2){

x = rand()%350;

a = 1+rand()%10;

printf("Х в градусах = %f\na = %d\n------------------------------------------\n", x, a);

}

else if (choose == 15){

printf(RED"\n------------------------------------------\n 23.11.21 - the day I bought a new laptop\n------------------------------------------\n\n"RESET);

goto start;

}

else {

printf("Error...Невідоме значення.");

goto start;

}

}

//////////////////////////////////////////////////////////////////////////////////

printf("Обчислення радіанів\n");

r = (x/180)\*Pi;

printf("Рaдіан = <%f>\n------------------------------------------\n", r);

//////////////////////////////////////////////////////////////////////////////////

printf("Стандартна точність обчислення(значення яке дорівнює кількость елементів суми): 10 .\nБажаєте змінити?\n"BLUE"<1>"RESET" - так.\n"BLUE"<0>"RESET" - ні.\nВибір: ");

scanf("%d", &choose);

if (choose == 1){

printf("Введіть точність: ");

scanf("%d", &t);

}

else t = 10;

printf("------------------------------------------\n");

//////////////////////////////////////////////////////////////////////////////////

double p1,p2,p12,p22,e1,e2;

e1 = 0;

e2 = 0;

p1 = a\*sin(2\*r);

p2 = a\*cos(2\*r);

for(int k = 0;k < t;k++)

{

p12 = p12 + a\*(pow(-1,k)\*(pow(2,1+2\*k))\*(pow(r,2\*k+1)/(factorial(2\*k +1))));

p22 = p22 + a\*(pow(-4,k)\*(pow(r,2\*k)/(factorial(2\*k))));

}

e1 = p12 - p1;

e2 = p22 - p2;

//////////////////////////////////////////////////////////////////////////////////

printf(RED"a\*sin(2\*r)\t\t\ta\*cos(2\*x)"RESET"\np1 = %lf\t\tp2 = %lf\np12 = %lf\t\tp22 = %lf\n\ne1 = %.7lf\t\te2 = %.7lf", p1,p2,p12,p22,e1,e2);

}

/////////////////////////////////////////

double factorial(double n){

if(n <= 1){

return 1;

}

else{

return (n\*factorial(n-1));

}

}