Міністерство освіти і науки України

Інститут телекомунікацій, радіоелектроніки та електронної техніки

**Звіт до лабораторної роботи № 13-14**

# Тема: «Структура функції. Локальні та глобальні змінні. Класи пам’яті»

з дисципліни «Програмування частина 2»

Варіант № 6

виконав студент групи АП-11

Головацький Назар

перевірив доцент кафедри ТК

Чайковський І.Б

Львів 2024

**Мета роботи:** навчитися використовувати функції у процесі програмування, розуміти особливості використання локальних та глобальних змінних та специфікаторів різних класів пам’яті.

**Завдання:**

2. Здійснити виконання прикладів, представлених у теоретичних відомостях, після чого представити скріни їх коду та результати виконання у звіті.

1)

#include<stdio.h>

#include<math.h>

double Geron (double a,double b,double c); /\*оголошення функцiї- прототипу\*/

double Geron (double a, double b, double c) { /\*реалізація функції\*/

double p; /\* локальна змiнна \*/

p= (a + b + c)/2;

return sqrt(p\*(p-a)\*(p-b)\*(p-c));/\*обчислити вираз i повернути результат\*/

}

int main ( ) {

double u, v, w;

double s;

printf ("Vvedit storonu trikutnika"); /\* у консолі введіть значення 2, 3, 4 \*/

scanf ("%lf %lf %lf",&u, &v,&w);

s=Geron(u,v,w); /\*викликається функцiя Geron. До аргументiв a,b,c

передаються значення змiнних u,v,w. Результат виклику присвоюється змiннiй s\*/

printf("Ploscha 1 trikutnika %lf\n",s);

s=Geron (10.3,8.1,9.7); /\* викликається функцiя Geron. До аргументiв a,b,c

передаються константи \*/

printf("Ploscha 2 trikutnika %lf\n",s);

s=Geron(u+10.3,v+w,w\*1.7); /\* викликається функцiя Geron. До аргументiв

передаються значення виразiв \*/

printf("Ploscha 3 trikutnika %lf\n",s);

return 0;

}

**Результат:**

Vvedit storonu trikutnika4

4

7

Ploscha 1 trikutnika 6.777721

Ploscha 2 trikutnika 36.928095

Ploscha 3 trikutnika 63.816781

2)

#include<stdio.h>

#include<math.h>

double fract(int,int);

double factorial(int);

int main( ) {

int m,n,s;

scanf ("%d %d",&m,&n);

printf ("%lf\n",fract(m+1,n+1));

return 0;

}

double fract(int x, int y) {

double t;

t=factorial(x+y)/(x\*y);

return t;

}

double factorial (int n) {

int i;

double p;

p=1;

for(i=1;i<=n;++i)

p\*=i;

return p;

}

**Результат:**

2

2

80.000000

3. Написати програму з використанням функції, яка друкує визначену кількість символів рядка. Уточнення: дана функція повинна приймати рядок символів і ціле число, яке визначатиме кількість символів, що слід надрукувати. Скрін коду програми та результати її виконання представити у звіті.

#include <stdio.h>

// Функція для друку визначеної кількості символів рядка

void printLimitedChars(const char \*str, int limit) {

int i;

// Проходимося по кожному символу рядка до досягнення ліміту або до завершення рядка

for (i = 0; i < limit && str[i] != '\0'; i++) {

putchar(str[i]); // Друкуємо поточний символ

}

putchar('\n'); // Друкуємо символ нового рядка після виводу лімітованої кількості символів

}

int main() {

char str[] = "Це рядок для тестування функції";

int limit = 10; // Кількість символів для друку

printf("Лімітований вивід рядка:\n");

printLimitedChars(str, limit);

return 0;

}

**Результат:**

**1)** **int limit = 10**

Лімітований вивід рядка:

Це ряд

**2) int limit = 20**

Лімітований вивід рядка:

Це рядок дл

**Висновок:** Я вивчив Структурні функції. Локальні та глобальні змінні. Класи пам’яті