**ЛАБОРАТОРНА РОБОТА № 18**

**Функції**

**Мета:** отримати практичні навички написання процедур і функцій за допомогою конструкцій мови, а також вибору правильного способу передачі параметрів.

**Хід роботи:**

**Завдання 1.**



Лістинг програми:

#include <iostream>

#include <math.h>

#include <windows.h>

int rectangleS(int a, int b) {

int S;

S = a \* b;

return S;

}

void main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

printf("Написати функцію обчислення площі прямокутника:\n");

int a, b;

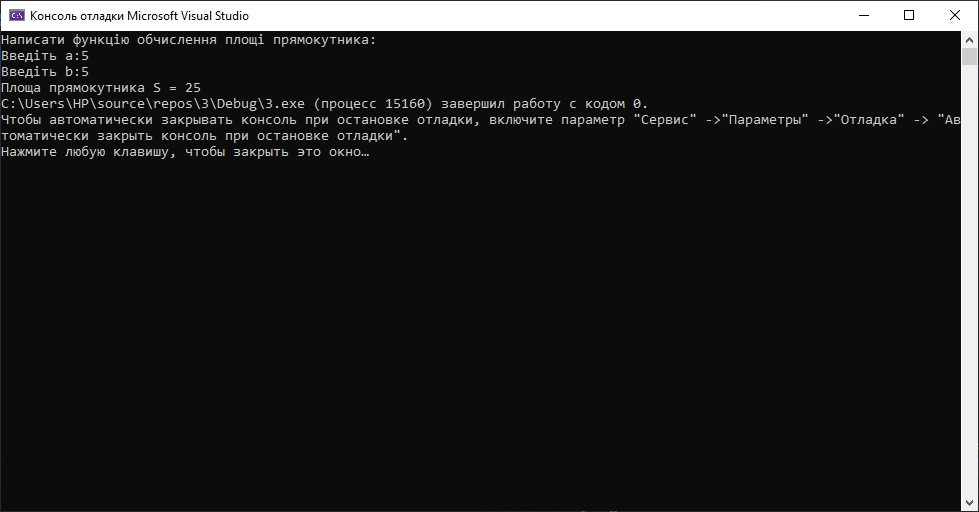
printf("Введіть а:"); scanf\_s("%d", &a);

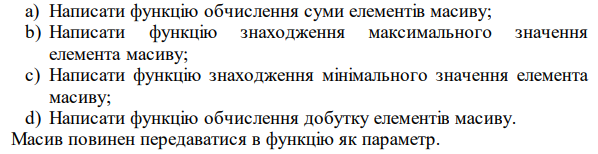
printf("Введіть b:"); scanf\_s("%d", &b);

printf("Площа прямокутника S = %d", rectangleS(a, b));

}

Результат програми:



**Завдання 2.** Дано масив. Використовуючи функції, вирішити наступні завдання: 

Лістинг програми:

#include <iostream>

#include <math.h>

#include <windows.h>

#include <time.h>

int sumARR(int arr[], int count) {

// Функція а)

int i, sumARR = 0;

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

sumARR += arr[i];

}

return sumARR;

}

int MAX(int arr[], int count) {

// Функція b)

int i, max = arr[0];

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

if (max < arr[i]) {

max = arr[i];

}

}

return max;

}

int MIN(int arr[], int count) {

// Функція c)

int i, min = arr[0];

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

if (min > arr[i]) {

min = arr[i];

}

}

return min;

}

int productARR(int arr[], int count) {

// Функція d)

int i, product = 1;

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

product \*= arr[i];

}

return product;

}

void main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

// Завдання

printf("Дано масив. Використовуючи функції, вирішити наступні завдання:\na) Написати функцію обчислення суми елементів масиву;\nb) Написати функцію знаходження максимального значення елемента масиву;\nc) Написати функцію знаходження мінімального значення елемента масиву;\nd) Написати функцію обчислення добутку елементів масиву.\nМасив повинен передаватися в функцію як параметр.\n");

int arr[100], a, b, count;

// Введення даних

printf("Введіть цілий розмір масиву:"); scanf\_s("%d", &count);

printf("Введіть початкове значення масиву a (ціле):"); scanf\_s("%d", &a);

printf("Введіть кінцеве значення масиву b (ціле):"); scanf\_s("%d", &b);

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

arr[i] = a + rand() % (b - a + 1);

printf("%d ", arr[i]);

}

// a)

sumARR(arr, count);

printf("\na) Сума елементів масиву = %d", sumARR(arr,count));

// b)

MAX(arr, count);

printf("\nb) Максимальне значення елементу масиву = %d", MAX(arr, count));

// c)

MIN(arr, count);

printf("\nc) Мінімальне значення елементу масиву = %d", MIN(arr, count));

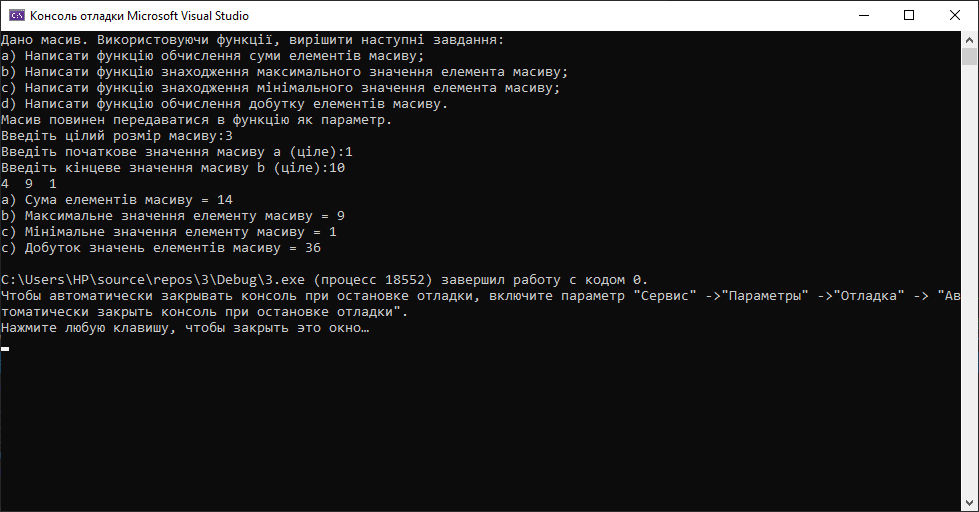
// d)

productARR(arr, count);

printf("\nc) Добуток значень елементів масиву = %d\n", productARR(arr, count));

}

Результат програми:



**САМОСТІЙНА РОБОТА № 18**

1. Запишіть прототип функції, яка приймає два цілочисельних аргументу і повертає дійсне число.

Лістинг програми:

#include <iostream>

#include <math.h>

#include <windows.h>

#include <time.h>

float backREALA(int first) {

float a = int (first);

return a;

}

float backREALB(int second) {

float b = int(second);

return b;

}

void main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

// Завдання

printf("Запишіть прототип функції, яка приймає два цілочисельних аргумента і повертає дійсне число.\n");

int a, b;

// Введення даних

printf("Введіть значення a (ціле):"); scanf\_s("%d", &a);

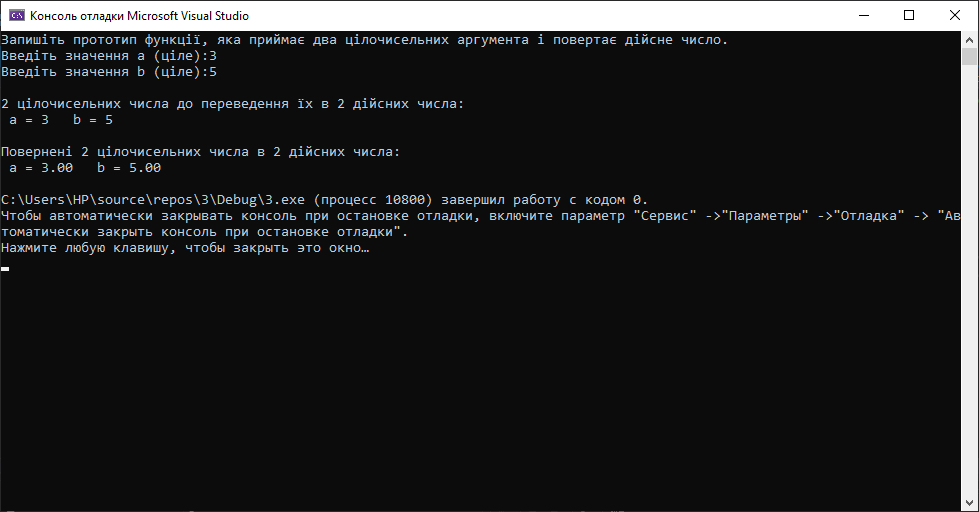
printf("Введіть значення b (ціле):"); scanf\_s("%d", &b);

printf("\n2 цілочисельних числа до переведення їх в 2 дійсних числа:\n a = %d b = %d\n", a, b);

printf("\nПовернені 2 цілочисельних числа в 2 дійсних числа:\n a = %.2f b = %.2f\n", backREALA(a), backREALB(b));

}

Результат програми:



1. Припустимо, дані три функції: int abs(int x); float abs(float x); long abs(long x). Яка з цих трьох функцій буде викликана в рядку float a = abs(-6);?

Відповідь: У рядок типу float буде заходити лише функція типу float, в іншому випадку буде чи виводитись значення не того формату, чи взагалі буде виводити 0.

#include <iostream>

#include <math.h>

#include <windows.h>

#include <time.h>

int abs1(int xx) {

int rezult;

rezult = abs(xx);

return rezult;

}

float abs2(float x1) {

float rezult;

rezult = abs(x1);

return rezult;

}

long abs3(long x1) {

long rezult;

rezult = abs(x1);

return rezult;

}

void main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

// Завдання

printf("Припустимо, дані три функції: int abs(int x); float abs(float x); long abs(long x).\nЯка з цих трьох функцій буде викликана в рядку float a = abs(-6);?\n");

int x = -6;

float a = abs1(x);

printf("\nРядок float a = abs(-6) буде викликана функція:%.1f\n", a);

}

1. Напишіть функцію зведення числа до квадрату. Напишіть функцію, що отримує ціле значення і повертає число з оберненим порядком цифр. Наприклад, для 7631 функція повинна повернути 1367.

Лістинг програми:

#include <iostream>

#include <math.h>

#include <windows.h>

int numSQUARE(int number) {

int square;

square = pow(number, 2);

return square;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

printf("Напишіть функцію зведення числа до квадрату.\n");

int num, squareNUM;

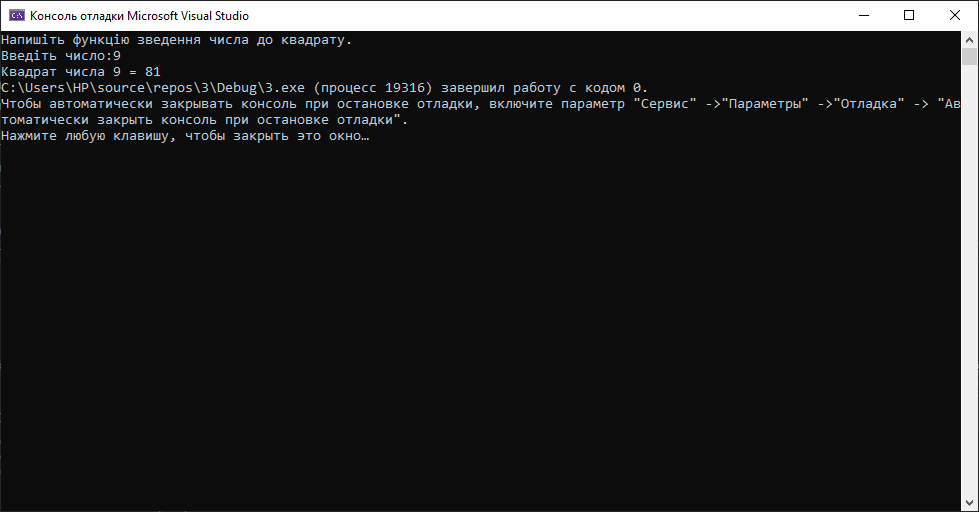
printf("Введіть число:"); scanf\_s("%d", &num);

squareNUM = numSQUARE(num);

printf("Квадрат числа %d = %d", num, squareNUM);

}

Результат програми:



1. Напишіть функцію, що отримує ціле значення і повертає число з оберненим порядком цифр. Наприклад, для 7631 функція повинна повернути 1367.

Лістинг програми:

#include <iostream>

#include <math.h>

#include <windows.h>

int rotateNUM(int number) {

// Функція перевертання числа навпаки

int i, temp, newNUM = 0;

while (number > 0) {

temp = number % 10; // знаходимо залишок, останню цифру

number = number / 10; // ділим націло, тобто удаляємо останню цифру

newNUM = newNUM \* 10; // збільшуємо розрядність другого числа

newNUM = newNUM + temp; // додаємо цифру

}

return newNUM;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

printf("4. Напишіть функцію, що отримує ціле значення і повертає число з оберненим порядком цифр.\nНаприклад, для 7631 функція повинна повернути 1367.\n");

int number, rotatedNUM, choose;

printf("Введіть 1(почати), 0(закінчити програму):"); scanf\_s("%d", &choose);

do {

if (choose == 0) {

break;

}

printf("Введіть число:"); scanf\_s("%d", &number);

rotatedNUM = rotateNUM(number);

printf("Число %d з оберненим порядком цифр = %d\n", number, rotateNUM(number));

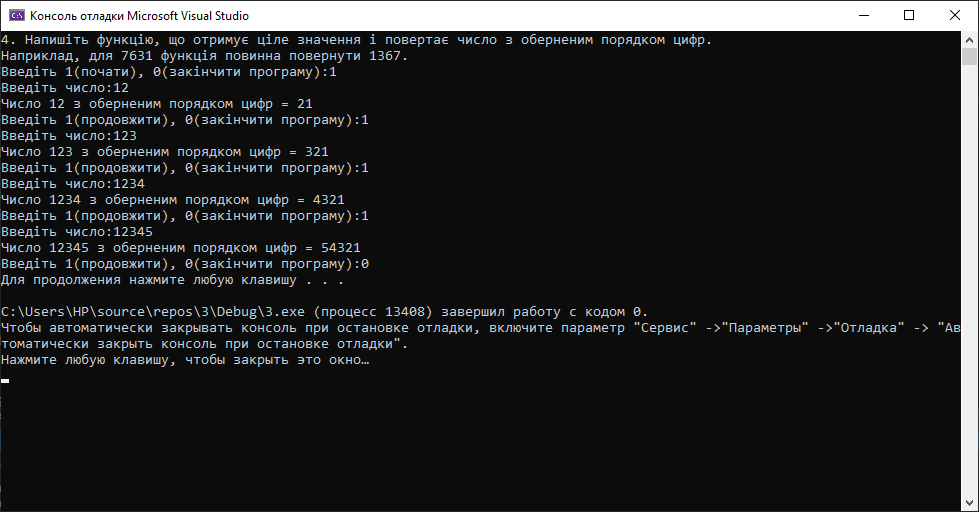
printf("Введіть 1(продовжити), 0(закінчити програму):"); scanf\_s("%d", &choose);

} while (choose);

system("pause");

}

Результат програми:



***Висновки:*** в ході виконання лабораторної роботи було ознайомлено з середовищем MS Visual Studio. Досліджено та отримано практичні навики щодо створення програм.