**ЛАБОРАТОРНА РОБОТА № 17**

**Робота з двовимірними масивами**

**Мета:** набуття практичних навичок роботи з двовимірними масивами.

**Хід роботи:**

**Завдання 1.** Дана квадратна матриця.



Лістинг програми:

#include <iostream>

#include <math.h>

#include <time.h>

#include <windows.h>

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

int a, b, i0, j0, arr1[100][100], arr2[100], size2, fl, temp;

printf("Дана квадратна матриця.\nРозмістити елементи парних рядків у порядку зростання.\n");

printf("Введіть кількість рядочків:"); scanf\_s("%d", &i0);

printf("Введіть кількість стовпчиків:"); scanf\_s("%d", &j0);

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

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

//Виведення початкового двовимірного масиву:

printf("Двовимірний масив(до сортування):\n");

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

arr1[i][j] = a + rand() % (b - a + 1);

printf("%3d", arr1[i][j]);

}

printf("\n");

}

// Перехід 2-ух вимірного масиву в одновимірний, але лише парні рядки.

size2 = 0;

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

if (i % 2 == 0) {

arr2[size2] = arr1[i][j];

size2++;

}

}

}

// Вивід одновимимірного масиву:

printf("Одновимірний масив(до сортування):\n");

for (int i = 0; i < size2; i++) {

printf("%d ", arr2[i]);

}

// Одновимірний масив після сортування:

do

{

fl = 0;

for (int i = 1; i < size2; i++) // Проходимо по кожному елементу у масиві

if (arr2[i - 1] > arr2[i]) // Ящко елемент який стоїть перед і-тим елементом масиву більше за і-того, робимо наступний код

{

temp = arr2[i]; // Присвоюємо тимча-совій змінній значення елемента i масиву

arr2[i] = arr2[i - 1]; // Присвоюємо змінній i масиву значення елемента стоячого до нього

arr2[i - 1] = temp; // Значення елемента і при-своюємо елементу стоячого до нього(тобто змінюємо іх місцями у масиві)

fl = 1; // Прапор для завершення циклу коли він підійде до кінця

}

} while (fl);

printf("\nОдновимірний масив пісня сортування:\n");

for (int i = 0; i < size2; i++) {

printf("%d ", arr2[i]);

}

// Підстановка одновимірного масиву у двовимірний:

size2 = 0;

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

if (i % 2 == 0) {

arr1[i][j] = arr2[size2];

size2++;

}

}

}

printf("\nДвовимірний масив після сортування парних рядків:\n");

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

printf("%3d", arr1[i][j]);

}

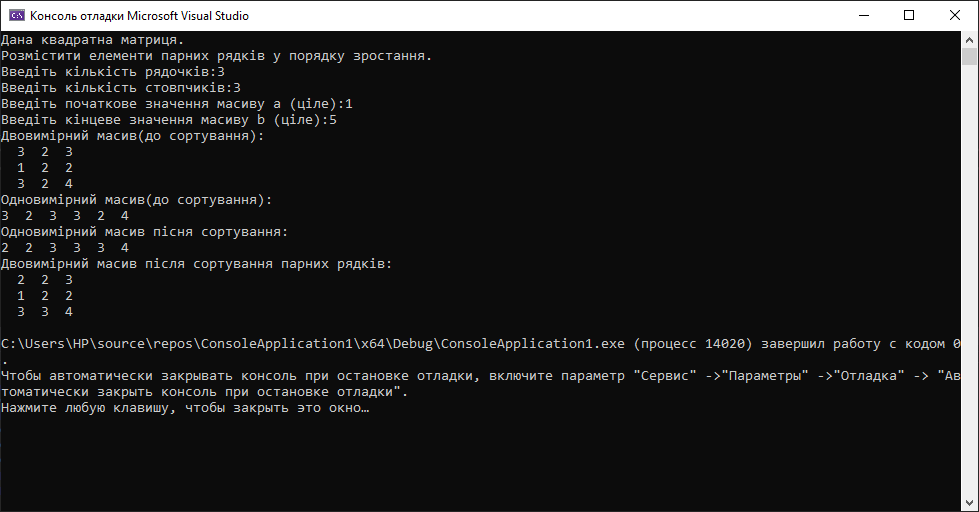
printf("\n");

}

return 0;

}

Результат програми:



**Завдання 2.** Дана прямокутна матриця.



Лістинг програми:

#include <iostream>

#include <math.h>

#include <time.h>

#include <windows.h>

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

int a, b, i0, j0, arr1[100][100], size2, fl, temp, last, k, n;

// Введнення даних:

printf("Циклічно зсунути парні рядки матриці зліва направо на k позицій.\n");

printf("Введіть кількість рядочків:"); scanf\_s("%d", &i0);

printf("Введіть кількість стовпчиків:"); scanf\_s("%d", &j0);

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

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

printf("Введіть значення k - зсув масиву на k позицій(ціле):"); scanf\_s("%d", &k);

// Виведення початкового двовимірного масиву:

printf("Двовимірний масив(до сдвигу):\n");

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

arr1[i][j] = a + rand() % (b - a + 1);

printf("%3d", arr1[i][j]);

}

printf("\n");

}

// Зсування масиву на k позицій вправо:

for (int n = 1; n <= k; n++) {

for (int i = i0 - 1; i >= 0; i--) {

for (int j = j0; j >= 0; j--) {

if (i % 2 == 0) {

if (j == j0 - 1) {

last = arr1[i][j];

}

arr1[i][j] = arr1[i][j - 1];

if (j == 0) {

arr1[i][j] = last;

}

}

}

}

}

// Вивід індексів двовимірного масиву:

printf("\nІндекси двовимірного масиву(для себе):\n");

for (int i = i0; i > -1; i--) {

for (int j = j0; j > -1; j--) {

printf("(%d-%d) ", i, j);

}

printf("\n");

}

// Вивід двовимірного масиву після зсування рядків на k позицій:

printf("\nДвовимірний масив після зсування рядків на k позицій:\n");

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

printf("%3d", arr1[i][j]);

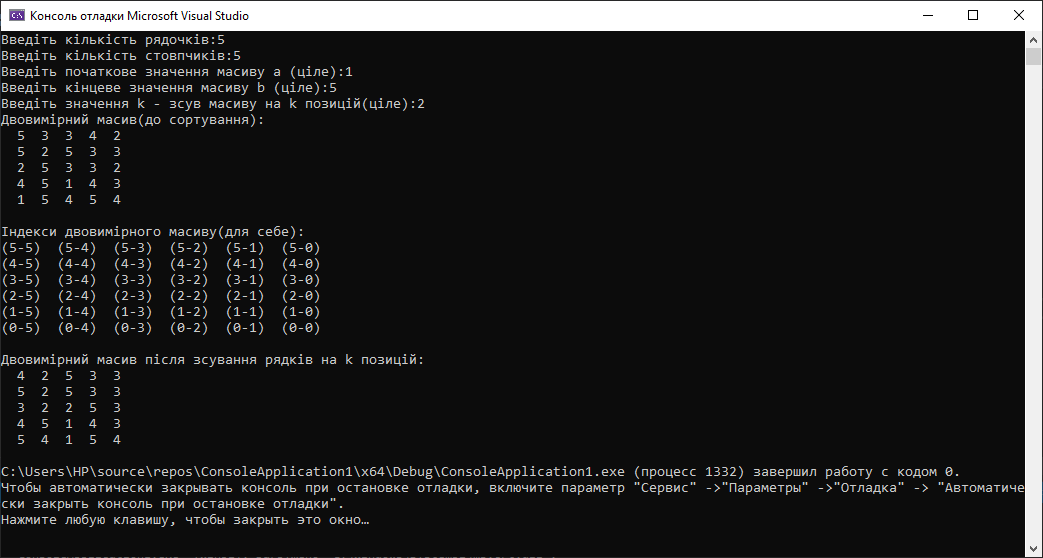
}

printf("\n");

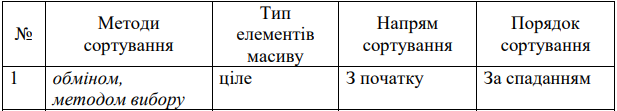
}

}

Результат програми:



**Завдання 3.** Здійснити сортування двовимірного масиву з наступними параметрами:



Лістинг програми:

#include <iostream>

#include <math.h>

#include <time.h>

#include <windows.h>

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

int a, b, i, j, i0, j0, arr1[100][100], arr2[100], fl, temp1, temp2, method, size2, iMIN;

// Введнення даних:

printf("Здійснити сортування двовимірного масиву з наступними параметрами:\nМетоди сортування: обміном, методом вибору.\nТип елементів масиву: ціле.\nНапрям сортування: з початку.\nПорядок сортування: за спаданням.\n");

printf("Введіть 1, якощо хочете сортувати масив методом обміну,\nчи 2, якщо хочете сортувати масив методом вибору:"); scanf\_s("%d", &method);

printf("Введіть кількість рядочків:"); scanf\_s("%d", &i0);

printf("Введіть кількість стовпчиків:"); scanf\_s("%d", &j0);

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

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

// Виведення початкового двовимірного масиву:

printf("Двовимірний масив(до сортування):\n");

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

arr1[i][j] = a + rand() % (b - a + 1);

printf("%3d", arr1[i][j]);

}

printf("\n");

}

// Методи сортування:

switch (method) {

case 1: // Сортування масиву методом обміну:

// Перетворення 2-овимірного масиву в 1-овимірний:

size2 = 0;

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

arr2[size2] = arr1[i][j];

size2++;

}

}

// Вивід одновимимірного масиву:

printf("Одновимірний масив(до сортування):\n");

for (int i = 0; i < size2; i++) {

printf("%d ", arr2[i]);

}

do

{

fl = 0;

for (int i = 1; i < size2; i++)

if (arr2[i - 1] < arr2[i]) {

temp1 = arr2[i];

arr2[i] = arr2[i - 1];

arr2[i - 1] = temp1;

fl = 1;

}

} while (fl);

// Вивід одновимірного масиву після сортування:

printf("\nОдновимірний масив(пісня сортування):\n");

for (int i = 0; i < size2; i++) {

printf("%d ", arr2[i]);

}

// Підстановка одновимірного масиву у двовимірний:

size2 = 0;

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

arr1[i][j] = arr2[size2];

size2++;

}

}

printf("\nДвовимірний масив(після сортування):\n");

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

printf("%3d", arr1[i][j]);

}

printf("\n");

}

break;

case 2: // Сортування масиву методом вибору:

// Перетворення 2-овимірного масиву в 1-овимірний:

size2 = 0;

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

arr2[size2] = arr1[i][j];

size2++;

}

}

// Вивід одновимимірного масиву:

printf("Одновимірний масив(до сортування):\n");

for (int i = 0; i < size2; i++) {

printf("%d ", arr2[i]);

}

for (int i = 0; i < size2 - 1; i++)

{

iMIN = i;

for (int j = i + 1; j < size2; j++)

if (arr2[j] > arr2[iMIN]) iMIN = j;

temp2 = arr2[i];

arr2[i] = arr2[iMIN];

arr2[iMIN] = temp2;

}

// Вивід одновимірного масиву після сортування:

printf("\nОдновимірний масив(пісня сортування):\n");

for (int i = 0; i < size2; i++) {

printf("%d ", arr2[i]);

}

// Підстановка одновимірного масиву у двовимірний:

size2 = 0;

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

arr1[i][j] = arr2[size2];

size2++;

}

}

printf("\nДвовимірний масив(після сортування):\n");

for (int i = 0; i < i0; i++) {

for (int j = 0; j < j0; j++) {

printf("%3d", arr1[i][j]);

}

printf("\n");

}

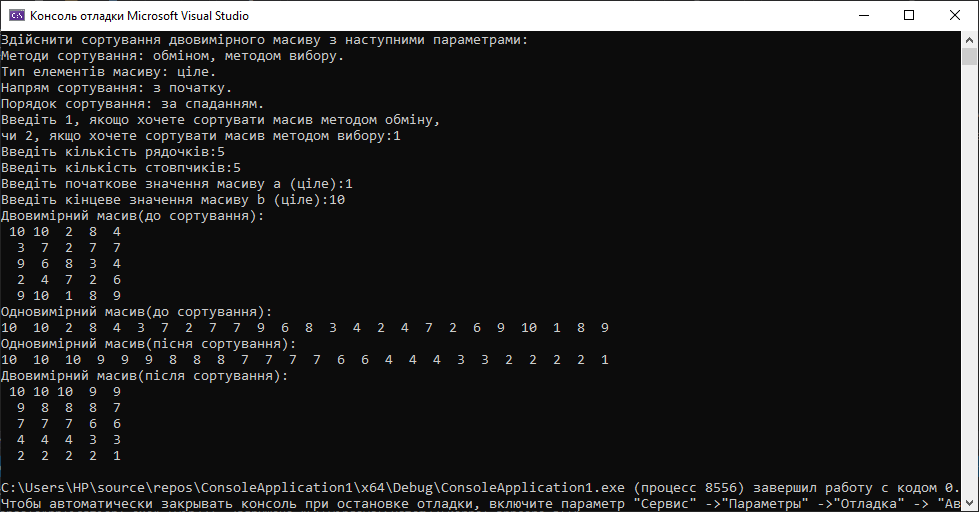
break;

}

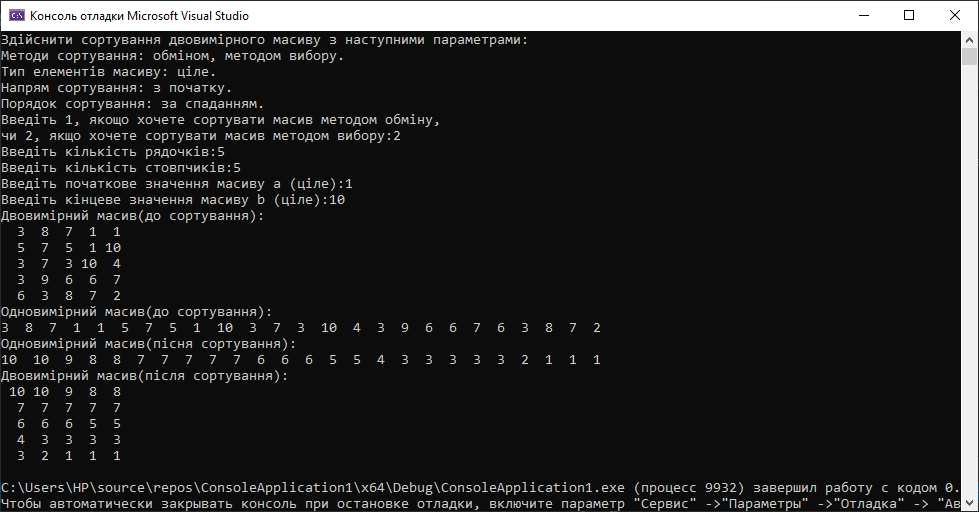
}

Результат програми:

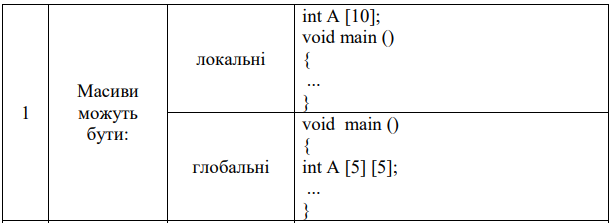
1. Сортування методом обміну:



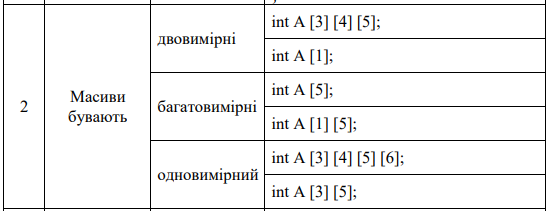
1. Сортування методом вибору:



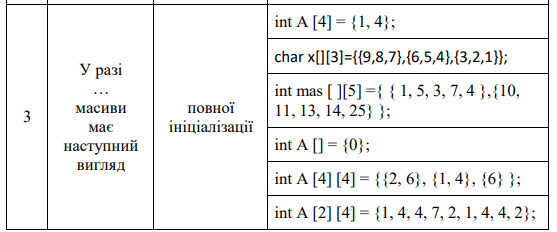
**Завдання 4.** Встановіть відповідність:

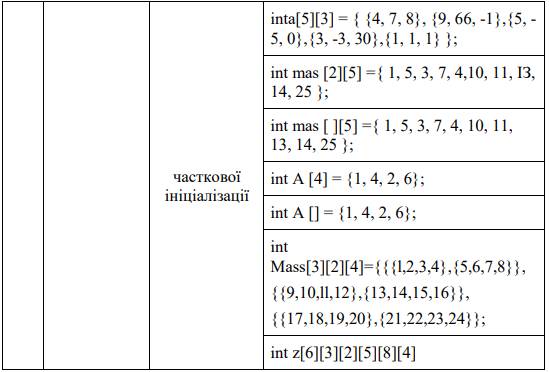
****

|  |  |  |
| --- | --- | --- |
| 1) | Локальні | Глобальні |
| Масив | void main(){  int A[5][5];  ...  } | int A[10];  void main(){  ...  } |



|  |  |  |  |
| --- | --- | --- | --- |
| 2) | одновимірні | двовимірні | багатовимірні |
| Масиви бувають | int A[1];  int A[5]; | int A[1][5];  int A[3][5]; | int A[3][4][5];  int A[3][4][5][6]; |

****

****

|  |  |  |
| --- | --- | --- |
| 3) | Повна ініціалізація | Часткова ініціалізація |
| У разі … масиви має наступний вигляд | char x[][3] = { {9,8,7},{6,5,4},{3,2,1} };  int a[5][3] = { {4, 7, 8}, {9, 66, -1}, {5, 5, 0}, {3, -3, 30}, {1, 1, 1} };  int mas[2][5] = { 1, 5, 3, 7, 4, 10, 11, 13, 14, 25 };  int mas[][5] = { { 1, 5, 3, 7, 4 },{10, 11, 13, 14, 25} };  int mas[][5] = { 1, 5, 3, 7, 4, 10, 11, 13, 14, 25 };  int A[4] = { 1, 4, 2, 6 };  int Mass[3][2][4] = { {{1,2,3,4},{5,6,7,8}}, {{9,10,11,12},{13,14,15,16}}, {{17,18,19,20},{21,22,23,24}} };  int z[6][3][2][5][8][4]; | int A[4] = { 1, 4 };  int A[] = { 0 }; (не можлива ініціалізація)  int A[4][4] = { {2, 6}, {1, 4}, {6} };  int A[2][4] = { 1, 4, 4, 7, 2, 1, 4, 4, 2 };  int A[] = { 1, 4, 2, 6 }; (не можлива ініціалізація) |

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