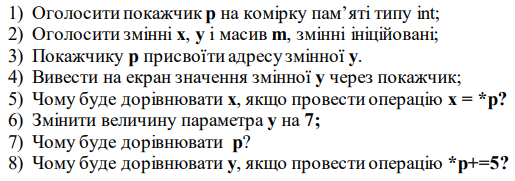
**ЛАБОРАТОРНА РОБОТА № 21**

**Покажчики**

**Мета:** набути навичок роботи з покажчиками в одновимірному масиві

**Хід роботи:**

**Завдання 1.** Написати програму з використанням покажчиків.



Лістинг программи:

#include <iostream>

#include <math.h>

#include <windows.h>

#include <time.h>

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

// Тема завдання

printf("Завдання 1:\nНаписати програму з використанням покажчиків.\n");

// 1)

int\* p;

// 2)

int x = 1, y = 9, arr[5] = { 1,3,5,7,9 };

// 3)

p = &y; // р присвоїлось значення адресу y

// 4)

printf("y = %d\n", \*p); // виведення значення у через покажчик

// 5)

x = \*p; // х присвоїлось значення покажчика(тобто значення у через його адрес)

printf("x = %d\n", x);

// 6)

y = 7; // зі зміною значення змінної, міняється також значення покажчика \*р

// 7)

printf("y = %d, p = %d\n", y, p); // значення у змінилося(7), тобто покажчик буде показувати адресу змінної у

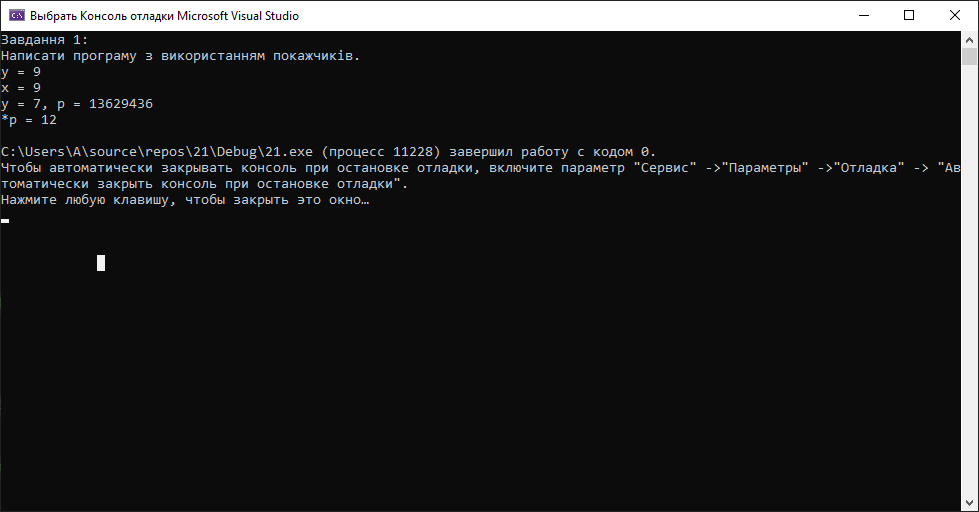
// 8)

\*p += 5; // значення у, яке передалось через покажчик, до нього додалось 5

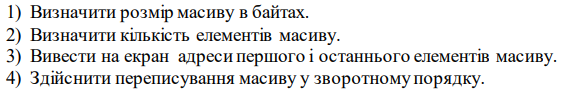
printf("\*p = %d\n", y, \*p); // тобто значення покажчика р буде = значенню у, тобто 7

}

Результат програми:



**Завдання 2.** Дано масив. Скласти програму де необхідно:



Лістинг программи:

#include <iostream>

#include <math.h>

#include <windows.h>

#include <time.h>

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

// Тема завдання

printf("Завдання 2:\nДано масив. Скласти програму де необхідно:\n1) Визначити розмір масиву в байтах.\n2) Визначити кількість елементів масиву.\n3) Вивести на екран адреси першого і останнього елементів масиву.\n4) Здійснити переписування масиву у зворотному порядку.\n");

int choise, count, a, b, arrA[69], arrB[69];

int\* pARRAFIRST, \* pARRALAST;

float sizeofARRA, sizeofELARRA;

printf("Почати програму(1) / Завершити програму(0):"); scanf\_s("%d", &choise);

do {

// 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);

printf("Масив БЕЗ використання показчика\*:\n");

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

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

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

}

printf("\nМасив методом з показчиком\*:\n");

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

\*(arrA+i) = a + rand() % (b - a + 1);

printf("%d ", \*(arrA+i));

}

// 1)

sizeofARRA = (sizeof arrA);

printf("\nРозмір масиву arrA в байтах = %.1f\n", sizeofARRA);

// 2)

sizeofELARRA = (sizeof arrA) / (sizeof arrA[0]);

printf("\nКількість елементів масиву arrA, на яких виділено пам'ять = %.1f\n", sizeofELARRA);

// 3)

pARRAFIRST = &arrA[0]; pARRALAST = &arrA[count - 1];

printf("\nАдрес елементу масиву arrA[0] = %d; Адрес елементу масиву arrA[%d] = %d.\n", pARRAFIRST, count - 1, pARRALAST);

// 4)

printf("\nМасив БЕЗ використання показчика\* у зворотньому порядку:\n");

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

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

}

printf("\nМасив методом з показчиком\* у зворотньому порядку:\n");

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

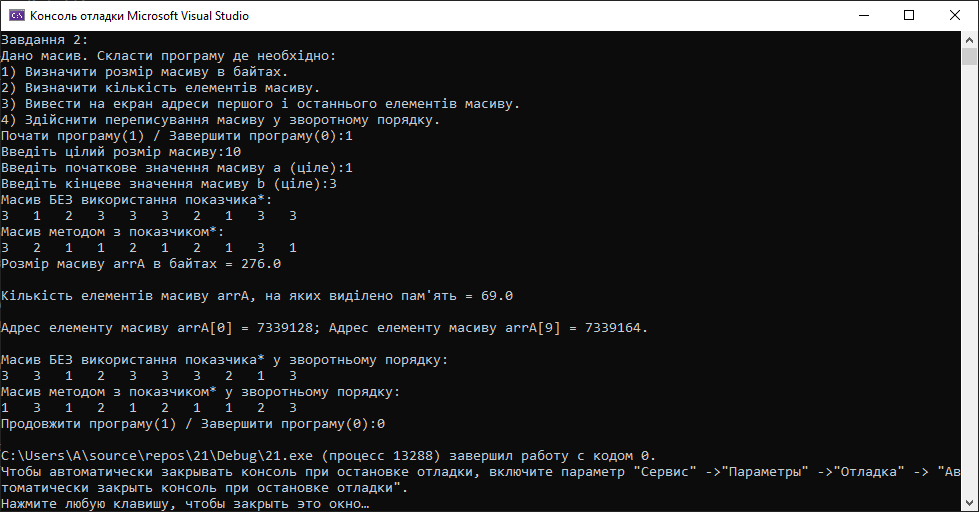
printf("%d ", \*(arrA + count - 1 - i));

}

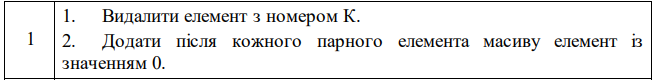
printf("\nПродовжити програму(1) / Завершити програму(0):"); scanf\_s("%d", &choise);

} while (choise);

Результат програми:



**Завдання 3.** Сформувати одновимірний масив цілих чисел, використовуючи датчик випадкових чисел.



**Не розумію чому воно виводить дублікати чисел та не пише 0 там де парні числа(3 години думав думав, ідей більше немає). Пише лише в випадку коли масив вже зрушений.**

Лістинг программи:

#include <iostream>

#include <math.h>

#include <windows.h>

#include <time.h>

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

// Тема завдання

printf("Завдання 3:\nСформувати одновимірний масив цілих чисел, використовуючи датчик випадкових чисел.\n1. Видалити елемент з номером К.\n2. Додати після кожного парного елемента масиву елемент із значенням 0.\n");

int choise, count, a, b, k, countPAIRED = 0, temp, last;

int\* arr1 = NULL; int\* arr2 = NULL;

printf("Почати програму(1) / Завершити програму(0):"); scanf\_s("%d", &choise);

do {

// 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);

printf("Введіть індекс елемента k, який хочете видалити:"); scanf\_s("%d", &k);

arr1 = (int\*)malloc(sizeof(int) \* count); // виділення памяті під потрібну кількість елементів у масиві задану користувачем

arr2 = arr1;

printf("\nСформований масив:\n");

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

{

\*(arr1 + i) = a + rand() % (b - a + 1);

printf("%d ", \*(arr1 + i));

}

// 1)

\*(arr1 + k) = 0; // видалення елементу під індексом k

for (int j = 1; j <= 1; j++) {

for (int i = k; i <= count - 1; i++) {

\*(arr1 + i) = \*(arr1 + i + 1);

}

}

printf("\nМасив з видаленим елементом під індексом k:\n");

for (int i = 0; i < count - 1; i++) printf("%d ", \*(arr1 + i));

// 2)

for (int i = 0; i < count - 1; i++) if (i % 2 == 0) countPAIRED++; // обрахунок кількості парних елементів

printf("\nНовий переформований масив зі вставленими нулями:\n");

int SIZENEW = count + countPAIRED;

arr2 = (int\*)realloc(arr1, SIZENEW \* sizeof(int)); // перевизначення потрібної памяті для масиву

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

if (i % 2 == 0) {

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

for (int l = SIZENEW - 1; l >= i; l--) {

if (l == SIZENEW - 1) {

last = 0;

}

arr2[l] = arr2[l - 1];

if (l == i) {

arr2[l] = 0;

}

}

}

printf("%d ", \*(arr2 + i));

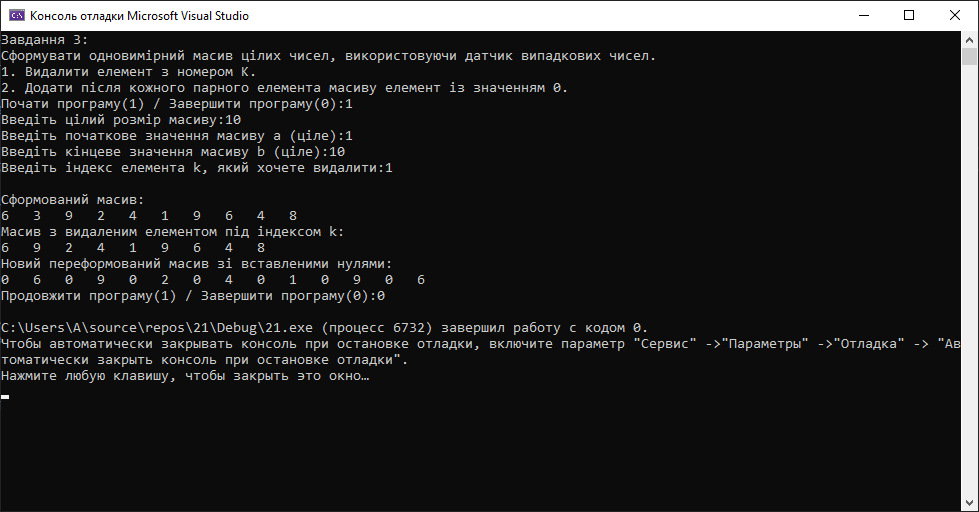
}

printf("\nПродовжити програму(1) / Завершити програму(0):"); scanf\_s("%d", &choise);

} while (choise);

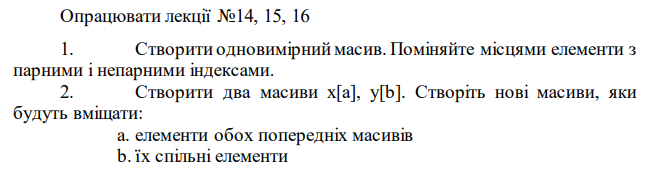
}

Результат програми:



**Самостійна робота**

**Завдання.**

****

Лістинг програми:

#include <iostream>

#include <math.h>

#include <windows.h>

#include <time.h>

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

// Тема завдання

printf("Самостійна робота:\n1. Створити одновимірний масив. Поміняйте місцями елементи з парними і непарними індексами.\n2. Створити два масиви x[a], y[b]. Створіть нові масиви, які будуть вміщати :\na.елементи обох попередніх масивів\nb.їх спільні елементи\n");

int choise, count, a, b, temp, countMEM = 0;

int\* x = NULL; int\* y = NULL; int\* arr = NULL; int\* z = NULL;

printf("Почати програму(1) / Завершити програму(0):"); scanf\_s("%d", &choise);

do {

// 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);

arr = (int\*)malloc(sizeof(int) \* count); // виділення памяті під потрібну кількість елементів у масиві задану користувачем

printf("\nСформований масив arr:\n");

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

{

\*(arr + i) = a + rand() % (b - a + 1);

printf("%d ", \*(arr + i));

}

// 1)

printf("\nМасив зі зміненими парними і непрарними елементами мясцями:\n");

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

if (i % 2 == 0) { // заміна парних елементів масиву непарними, а непарних - парними

temp = \*(arr + i);

\*(arr + i) = \*(arr + i + 1);

\*(arr + i + 1) = temp;

}

printf("%d ", \*(arr + i));

}

// 2)

// a)

x = (int\*)malloc(sizeof(int) \* count); // виділення памяті під потрібну кількість елементів у масиві задану користувачем

printf("\na) Сформований масив x:\n");

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

{

\*(x + i) = a + rand() % (b - a + 1);

printf("%d ", \*(x + i));

}

y = (int\*)malloc(sizeof(int) \* count); // виділення памяті під потрібну кількість елементів у масиві задану користувачем

printf("\nСформований масив y:\n");

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

{

\*(y + i) = a + rand() % (b - a + 1);

printf("%d ", \*(y + i));

if (\*(x + i) == \*(y + i)) countMEM++;

}

// b)

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

if (\*(x + i) == \*(y + i)) {

countMEM++; // обрахунок кількості однакових елементів масивів x and y

}

}

printf("\nb) Новий переформований масив зі спільними елементами масивів x and y:\n");

int SIZENEW = countMEM;

z = (int\*)realloc(x, SIZENEW \* sizeof(int)); // перезадання кількості пам'ять для масиву z

// вивід масиву зі спільними елементами

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

if (\*(x + i) == \*(y + i)) {

printf("%d ", \*(z + i));

}

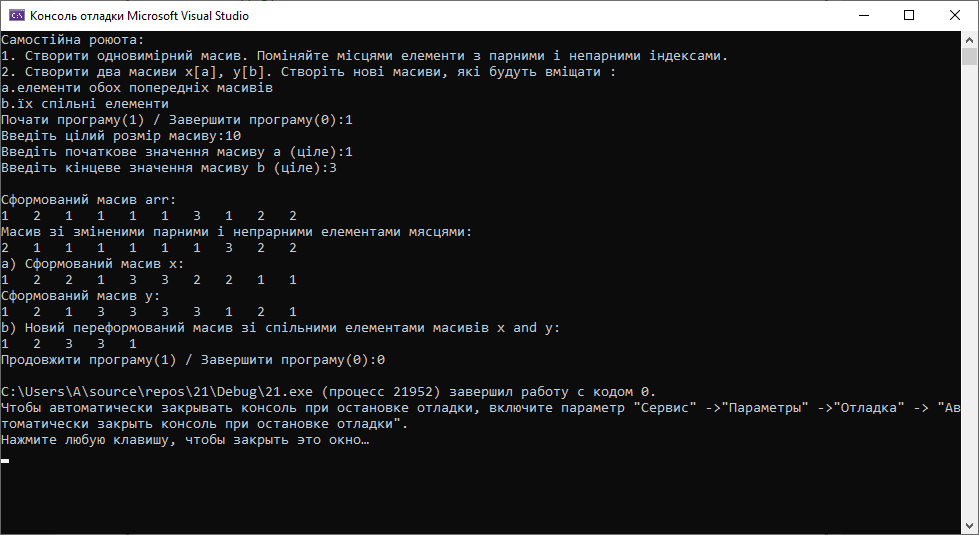
}

printf("\nПродовжити програму(1) / Завершити програму(0):"); scanf\_s("%d", &choise);

} while (choise);

}

Результат програми:



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