Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп’ютерних технологій, автоматики та метрології

Кафедра «Комп’ютеризовані системи автоматики»



ЗВІТ

про виконання лабораторної роботи №4

«МАСИВИ І ПОКАЖЧИКИ»

із навчальної дисципліни: «Алгоритми і структури даних»

Варіант №1

Виконав:

студент групи ІР-24

Кіндрат Віктор Романович

Прийняв:

к. ф.-м. н., доцент.

Сиротюк С. В.

Львів 2024

**Мета роботи** – ознайомитися з формами визначення одновимірних масивів у мові С, використанням покажчиків при роботі з масивами, навчитися формувати числові і символьні масиви.

**Програма 4\_1.**

У програмі визначається та ініціалізується одновимірний масив. Визначається мінімальний елемент масиву.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

int main()

{

int arr[10]={4,2,-3,8,12,1,-2,9,0,15}, min, i;

for(i=0; i<10; i++)

printf("arr[%d]=%3d %p\n",i,arr[i],&arr[i]);

printf("Size massiv arr: %d bait\n",sizeof(arr));

min=arr[0];

for(i=0; i<10; i++)

if(arr[i]<min) min=arr[i];

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

system("pause");

return 0;

}

**Завдання до програми 4\_1.**

Розробіть програму, в якій з клавіатури вводиться масив цілих чисел заданого розміру, виконуються задані обчислення. На екран виводяться значення елементів масиву і результати обчислень.

Обчисліть середнє арифметичне тільки ненульових елементів. Розмір масиву 6.

**Програмний код:**

#include <stdio.h>  
  
int main() {  
 int a[6];  
  
 for (int i = 0; i < 6; i++) {  
 scanf("%d", &a[i]);  
 }  
  
 int sum = 0;  
 int count\_of\_not\_null = 0;  
  
 for (int i = 0; i < 6; i++) {  
 if (a[i] != 0) {  
 count\_of\_not\_null++;  
 }  
 sum += a[i];  
 }  
  
 float median = sum / count\_of\_not\_null;  
  
 printf("Median of sequence: %.2f\n", median);  
 return 0;  
}

**Приклад виконання:**

viktorkindrat@MacBook-Air-Viktor lab4 % ./1.out

-> 10

-> 14

-> 12

-> 12

-> 21

-> 3

Median of sequence: 12.00

**Програма 4\_2.**

У програмі визначається і заповнюється випадковими числами одновимірний масив та підраховується кількість додатних чисел між мінімальним і максимальним елементами масиву.

#include <iostream>

using namespace std;

#include <time.h> // для запуску генератора випадкових чисел

#include <stdlib.h> // для підключення генератора випадкових чисел

int main()

{

const int n=20;

int a[n], i, imax=0, imin=0, kol=0;

srand((unsigned)time(NULL)); // запуск генератора випадк. чисел

for(i=0; i<n; i++)

{

a[i]=rand()%101-50; // значення від –50 до +50

cout<<" a["<<i<<"]="<<a[i]<<"\n";

}

for (i=0; i<n; i++) // знаходження max та min значення

{

if(a[i]>a[imax])imax=i;

if(a[i]<a[imin])imin=i;

}

cout<<" max a["<<imax<<"] = "<<a[imax];

cout<<"\n min a["<<imin<<"] = "<<a[imin];

int ibeg=imax<imin? imax : imin;

int iend=imax<imin? imin : imax;

for(i=ibeg+1; i<iend; i++) if(a[i]>0) kol++;

cout<<"\nBetween a["<<ibeg<<"] and a["<<iend<<"] ";

cout<<kol<<" positive value\n\n";

system("pause");

return 0;

}

**Завдання до програми 4\_2.**

Розробіть програму, яка виконує задані обчислення та виводить на екран масив і результат обчислення.

Заповніть масив цілих чисел випадковими числами від -40 до 40 та обчисліть кількість від’ємних чисел. Розмір масиву 15.

**Програмний код:**

#include <stdio.h> #include <stdlib.h> #include <time.h> // for random number generation  
  
int main() {  
 const int n = 15;  
 int a[n], i, count = 0;  
  
 srand((unsigned)time(NULL));  
  
 for (i = 0; i < n; i++) {  
 a[i] = rand() % 81 - 40;  
 printf("a[%d] = %d\n", i, a[i]);  
 }  
  
 for (i = 0; i < n; i++) {  
 if (a[i] < 0) {  
 count++;  
 }  
 }  
  
 printf("count of number less than 0 = %d\n", count);  
  
 return 0;  
}

**Приклад виконання:**

viktorkindrat@MacBook-Air-Viktor lab4 % ./2.out

a[0] = -31

a[1] = -25

a[2] = -12

a[3] = -22

a[4] = 39

a[5] = -35

a[6] = -6

a[7] = -34

a[8] = 14

a[9] = 24

a[10] = -6

a[11] = 11

a[12] = 38

a[13] = 30

a[14] = -22

count of number less than 0 = 9

**Програма 4\_3.**

В програмі визначається значення максимального елементу масиву, використовуючи метод доступу через розіменування покажчика.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

int main()

{

int arr[10]={5,3,2,-4,6,7,110,-17,0,13}, max, i=0;

int \*px= arr; // визначення покажчика типу int на масив

max=\*px; // привласнення max значення нульового елемента масиву

while(i<10)

{

if(\*(px+i)>max) max=\*(px+i);

i++;

}

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

system("pause");

return 0;

}

**Завдання до програми 4\_3.**

Розробіть програму, в якій визначається масив цілих чисел заданого розміру. Перший елемент масиву задається, всі інші елементи вводяться з клавіатури, потім виконуються задані обчислення, використовуючи метод доступу до елементів масиву через розіменування покажчика. На екран виводяться значення елементів масиву і результати обчислень.

Визначте мінімальний елемент масиву, що кратний 5. Перший елемент масиву дорівнює 500. Розмір масиву 8.

**Програмний код:**

#include <stdio.h> #include <stdlib.h>  
  
void fill\_array\_via\_int(int \*arr, int size) {  
 for (int i = 0; i < size; i++) {  
 puts("Enter an integer from your keyboard: ");  
 scanf("%d", &arr[i]);  
 }  
}  
  
int main() {  
 int arr[8] = {500};  
  
 fill\_array\_via\_int(&(arr[1]), 7);  
  
 int \*ptr = arr;  
 int min\_el\_that\_div\_5 = \*ptr;  
  
 for (int i = 1; i < 8; i++) {  
 if ((\*(ptr + i) < min\_el\_that\_div\_5) && ((\*(ptr + i)) % 5 == 0)) {  
 min\_el\_that\_div\_5 = \*(ptr + i);  
 }  
 }  
  
 printf("Minimum integer value that could be devided by 5 is %d\n", min\_el\_that\_div\_5);  
  
 return 0;  
}

**Приклад виконання:**

viktorkindrat@MacBook-Air-Viktor lab4 % ./3.out

Enter an integer from your keyboard:

10

Enter an integer from your keyboard:

15

Enter an integer from your keyboard:

1

Enter an integer from your keyboard:

2

Enter an integer from your keyboard:

3

Enter an integer from your keyboard:

4

Enter an integer from your keyboard:

5

Minimum integer value that could be devided by 5 is 5

**Програма 4\_4.**

Програма сортує елементи масиву методом вибору максимального елемента масиву.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

#define N 10

int main()

{

int dig[N]={50,3,25,4,6,7,11,17,0,13};

int i, j, mm, max, tmp;

puts("Old massiv");

for(i=0; i<N; i++)

printf("%4d ", dig[i]);

for(i=N-1; i>=1; i--)

{

max=dig[0]; mm=0;

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

if(dig[j]>max){max=dig[j]; mm=j;}

tmp=dig[i]; // запам’ятати останній елемент

dig[i]=max; // максимум на місце останнього елементу

dig[mm]=tmp; // останній елемент на місце максимуму

}

puts("\nNew massiv");

for(i=0; i<N; i++)

printf("%4d ", dig[i]);

system("pause");

return 0;

}

**Завдання до програми 4\_4.**

Розробіть програму, що сортує заданим методом одновимірний масив цілих чис

ел та виводить на екран масив до і після сортування.

Програму, що сортує масив методом бульбашка. Розмір масиву 10.

**Програмний код:**

#include <stdio.h> #include <stdlib.h> #include <time.h>  
  
#define SIZE 10  
  
void printArray(int arr[], int size) {  
 for (int i = 0; i < size; i++) {  
 printf("%d ", arr[i]);  
 }  
 printf("\n");  
}  
  
void bubbleSortDescending(int arr[], int size) {  
 for (int i = 0; i < size - 1; i++) {  
 for (int j = 0; j < size - i - 1; j++) {  
 if (arr[j] < arr[j + 1]) {  
 int temp = arr[j];  
 arr[j] = arr[j + 1];  
 arr[j + 1] = temp;  
 }  
 }  
 }  
}  
  
int main() {  
 int arr[SIZE];  
  
 srand(time(NULL));  
  
 for (int i = 0; i < SIZE; i++) {  
 arr[i] = rand() % 81 - 40;  
 }  
  
 printf("Масив до сортування:\n");  
 printArray(arr, SIZE);  
  
 bubbleSortDescending(arr, SIZE);  
  
 printf("Масив після сортування:\n");  
 printArray(arr, SIZE);  
  
 return 0;  
}

**Приклад виконання:**

viktorkindrat@MacBook-Air-Viktor lab4 % ./4.out

Масив до сортування:

-25 -35 -18 -37 24 35 29 -36 29 4

Масив після сортування:

35 29 29 24 4 -18 -25 -35 -36 -37

**Програма 4\_5.**

У програмі демонструється використання покажчиків при обробці символьних масивів.

#include <iostream>

using namespace std;

int main()

{

char str[50], \*p;

cout<<"Введіть рядок:\n";

cin.getline(str, 50);

p=str;

int count=0;

while(\*p!='\0')

{

if(\*p==' ') count++;

p++;

}

cout<<"Кількість пробілів: "<<count<<endl;

return 0;

}

**Завдання до програми 4\_5.**

Розробіть програму, що демонструє обробку символьних масивів з використанням покажчиків.

Програму, що обчислює довжину рядка.

**Програмний код:**

#include <stdio.h> #include <string.h>  
  
#define MAX\_LENGTH 256  
  
int main() {  
 char subject\_teacher[MAX\_LENGTH];  
 char message[] = " Консультація відбудеться в середу на п’ятій парі.";  
 char combined[MAX\_LENGTH];  
  
 printf("Введіть рядок з назвою предмету і прізвищем викладача: ");  
 fgets(subject\_teacher, sizeof(subject\_teacher), stdin);  
  
 subject\_teacher[strcspn(subject\_teacher, "\n")] = '\0';  
  
 int length\_subject\_teacher = strlen(subject\_teacher);  
 printf("Кількість символів у введеному рядку: %d\n", length\_subject\_teacher);  
  
 int length\_message = strlen(message);  
 printf("Кількість символів у повідомленні: %d\n", length\_message);  
  
 snprintf(combined, sizeof(combined), "%s%s", subject\_teacher, message);  
  
 printf("Сумарний рядок: %s\n", combined);  
 printf("Кількість символів у сумарному рядку: %d\n", strlen(combined));  
  
 return 0;  
}

**Приклад виконання:**

viktorkindrat@MacBook-Air-Viktor lab4 % ./4.out

Введіть рядок з назвою предмету і прізвищем викладача:   
-> Верес Командна робота в ІТ та презентаційні навички

Кількість символів у введеному рядку: 132

Кількість символів у повідомленні: 93

Сумарний рядок: В�ерес Алгоритмізація таКомандна ро�б�ота в ІТ та презента�ційні навички Консультація відбудеться в середу на п’ятій парі.

Кількість символів у сумарному рядку: 225

**Висновок:**

В ході виконання лабораторної роботи я опанував принципи роботи з масивами і покажчиками в мові програмування Сі. Виконані програми дозволяють виконувати різноманітні обчислення, використовуючи різні методи обробки масивів.