

**Міністерство освіти і науки України**  
**Львівський національний університет імені Івана Франка**  
**Факультет електроніки та комп'ютерних технологій**

**Звіт**  
про виконання лабораторної роботи №8  
“arrays”

Виконав:  
студент 1 курсу  
групи ФЕП-11  
Маначинський Тарас  
Викладач: Кужій Ю.

**Львів – 2024**

1. Написати функцію, яка повертає масив з 10 випадкових чисел з діапазону [0...99].

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  void findMinMaxAndSum(int arr[], int size, int* min, int* max, int* sum) {
6      *min = arr[0];
7      *max = arr[0];
8
9      for (int i = 1; i < size; i++) {
10         if (arr[i] < *min) {
11             *min = arr[i];
12         }
13         if (arr[i] > *max) {
14             *max = arr[i];
15         }
16     }
17
18     *sum = *min + *max;
19 }
20
21 int main() {
22     int arr[20];
23     int min, max, sum;
24
25     srand(time(NULL));
26
27
28     for (int i = 0; i < 20; i++) {
29         arr[i] = rand() % 100;
30         printf("%d ", arr[i]);
31     }
32     printf("\n");
33
34
35     findMinMaxAndSum(arr, 20, &min, &max, &sum);
36
37
38     printf("Мінімальний елемент: %d\n", min);
39     printf("Максимальний елемент: %d\n", max);
40     printf("Сума мінімального і максимального елементів: %d\n", sum);
41
42     return 0;
43 }
```

2.1 Написати функції, що повертають вказівник на елемент з довільним індексом одновимірного масиву з 10 елементів

```
1  #include <stdio.h>
2
3  int* getElementAtIndex(int arr[], int index) {
4      if (index >= 0 && index < 10) {
5          return &arr[index];
6      }
7      return NULL;
8  }
9
10 int main() {
11     int arr[10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};
12     int index = 5;
13     int* elementPtr = getElementAtIndex(arr, index);
14
15     if (elementPtr != NULL) {
16         printf("Елемент за індексом %d: %d\n", index, *elementPtr);
17     } else {
18         printf("Елемент з індексом %d не існує.\n", index);
19     }
20
21     return 0;
22 }
```

2.2 Написати функції, що повертають вказівник на елемент з довільними індексами двовимірного масиву розміром 12 x 12 елементів (повернути NULL, якщо елемент відсутній)

```
1  #include <stdio.h>
2
3  int* getElementAtIndices(int arr[12][12], int row, int col) {
4      if (row >= 0 && row < 12 && col >= 0 && col < 12) {
5          return &arr[row][col];
6      }
7      return NULL;
8  }
9
10 int main() {
11     int arr[12][12] = { {0} };
12     int row = 5, col = 7;
13     arr[row][col] = 42;
14
15     int* elementPtr = getElementAtIndices(arr, row, col);
16
17     if (elementPtr != NULL) {
18         printf("Елемент за індексами [%d][%d]: %d\n", row, col, *elementPtr);
19     } else {
20         printf("Елемент з індексами [%d][%d] не існує.\n", row, col);
21     }
22
23     return 0;
24 }
```

3 Написати функцію, котра в масиві з 20 випадкових чисел з діапазону [0...99] знаходить максимальний і мінімальний елементи та їхню суму

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  void findMinMaxAndSum(int arr[], int size, int* min, int* max, int* sum) {
6      *min = arr[0];
7      *max = arr[0];
8
9      for (int i = 1; i < size; i++) {
10         if (arr[i] < *min) {
11             *min = arr[i];
12         }
13         if (arr[i] > *max) {
14             *max = arr[i];
15         }
16     }
17
18     *sum = *min + *max;
19 }
20
21 int main() {
22     int arr[20];
23     int min, max, sum;
24
25     srand(time(NULL));
26
27
28     for (int i = 0; i < 20; i++) {
29         arr[i] = rand() % 100;
30         printf("%d ", arr[i]);
31     }
32     printf("\n");
33
34
35     findMinMaxAndSum(arr, 20, &min, &max, &sum);
36
37
38     printf("Мінімальний елемент: %d\n", min);
39     printf("Максимальний елемент: %d\n", max);
40     printf("Сума мінімального і максимального елементів: %d\n", sum);
41
42     return 0;
43 }
```