

1.array largest element c assi 1.cpp

```
1  #include <stdio.h>
2
3  int main() {
4      int arr[5] = {12, 56, 34, 78, 100};
5      int largest = arr[0];
6      for (int i = 1; i < 5; i++) {
7          if (arr[i] > largest) {
8              largest = arr[i];
9          }
10     }
11     printf("The largest element in the array is: %d", largest);
12     return 0;
13 }
```

C:\Users\91961\Documents\1.i × + - □ ×

The largest element in the array is: 100

Process exited after 4.171 seconds with return value 0
Press any key to continue . . . |

2.largest two numbers ass 2.cpp

```

1  #include <stdio.h>
2
3  int main()
4  {
5      int n, arr[100], first, second, i;
6      printf("Enter the number of elements: ");
7      scanf("%d", &n);
8
9      printf("Enter %d elements:\n", n);
10     for(i=0; i<n; i++)
11         scanf("%d", &arr[i]);
12
13     // Find the largest two numbers in the array
14     if(arr[0] > arr[1]) {
15         first = arr[0];
16         second = arr[1];
17     }
18     else {
19         first = arr[1];
20         second = arr[0];
21     }
22     for(i=2; i<n; i++) {
23         if(arr[i] > first) {
24             second = first;
25             first = arr[i];
26         }
27         else if(arr[i] > second && arr[i] != first) {
28             second = arr[i];
29         }
30     }
31
32     // Print the largest two numbers in the array
33     printf("The FIRST LARGEST = %d\n", first);
34     printf("THE SECOND LARGEST = %d\n", second);
35     return 0;

```

```

C:\Users\91961\Documents\2.  x + v - □ x
Enter the number of elements: 8
Enter 8 elements:
34 64 78 67 59 48 23 56
The FIRST LARGEST = 78
THE SECOND LARGEST = 67

-----
Process exited after 148.4 seconds with return value 0
Press any key to continue . . . |

```

3.one dimensional assi 2.cpp

```

1  #include <stdio.h>
2
3  int main() {
4      int n, arr[100], i, j, temp;
5      float avg, sum = 0;
6
7      // Read the number of elements
8      printf("Enter the number of elements: ");
9      scanf("%d", &n);
10
11     // Read the array elements
12     printf("Enter the array elements:\n");
13     for(i=0; i<n; i++) {
14         scanf("%d", &arr[i]);
15         sum += arr[i];
16     }
17
18     // Sort the array in descending order
19     for(i=0; i<n-1; i++) {
20         for(j=i+1; j<n; j++) {
21             if(arr[i] < arr[j]) {
22                 temp = arr[i];
23                 arr[i] = arr[j];
24                 arr[j] = temp;
25             }
26         }
27     }
28
29     // Find the second largest and smallest elements
30     int second_largest = arr[1];
31     int smallest = arr[n-1];
32     for(i=2; i<n; i++) {
33         if(arr[i] < smallest) {
34             smallest = arr[i];
35         }
36         else if(arr[i] > second_largest && arr[i] != arr[0]) {
37             second_largest = arr[i];
38         }
39     }
40
41     // Calculate the average of the second largest and smallest elements
42     avg = (second_largest + smallest) / 2.0;
43
44     // Check if the average is present in the array
45     for(i=0; i<n; i++) {
46         if(avg == arr[i]) {
47             printf("The average of the second largest and smallest elements is present in the array.\n");
48             return 0;
49         }
50     }
51     printf("The average of the second largest and smallest elements is not present in the array.\n");
52     return 0;
53 }

```

```

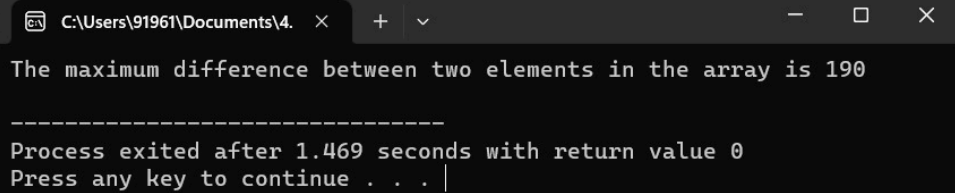
C:\Users\91961\Documents\3.
Enter the number of elements: 3
Enter the array elements:
2
3
4
The average of the second largest and smallest elements is not present in the array.

-----
Process exited after 90.4 seconds with return value 0
Press any key to continue . . .

```

4.maximum difference ass 2.cpp

```
1  #include <stdio.h>
2
3  int main() {
4      int array[] = {10, 15, 90, 200, 110};
5      int n = sizeof(array) / sizeof(int);
6      int max_diff = array[1] - array[0];
7      int min_elem = array[0];
8
9      for (int i = 1; i < n; i++) {
10         if (array[i] - min_elem > max_diff) {
11             max_diff = array[i] - min_elem;
12         }
13         if (array[i] < min_elem) {
14             min_elem = array[i];
15         }
16     }
17
18     printf("The maximum difference between two elements in the array is %d\n", max_diff);
19
20     return 0;
21 }
```



C:\Users\91961\Documents\4. × + ▾

The maximum difference between two elements in the array is 190

Process exited after 1.469 seconds with return value 0

Press any key to continue . . . |

5.duplicate elements ass 2.cpp

```

1  #include <stdio.h>
2
3  int main()
4  {
5      int arr[5] = {2, 7, 1, 23, 5};
6      int size = 5;
7      int i, j, k;
8
9      printf("Original Array: ");
10     for (i = 0; i < size; i++) {
11         printf("%d ", arr[i]);
12     }
13
14     // Remove duplicates
15     for (i = 0; i < size; i++) {
16         for (j = i+1; j < size; j++) {
17             if (arr[j] == arr[i]) {
18                 for (k = j; k < size; k++) {
19                     arr[k] = arr[k+1];
20                 }
21                 size--;
22             } else {
23                 j++;
24             }
25         }
26     }
27
28     printf("\nArray after removing duplicates: ");
29     for (i = 0; i < size; i++) {
30         printf("%d ", arr[i]);
31     }
32
33     return 0;

```

```

C:\Users\91961\Documents\5.
Original Array: 2 7 1 23 5
Array after removing duplicates: 2 7 1 23 5
-----
Process exited after 1.486 seconds with return value 0
Press any key to continue . . .

```

6.even assi 2.cpp

```
1 #include <stdio.h>
2
3 #define MAX_SIZE 100
4
5 int main()
6 {
7     int arr[MAX_SIZE], even[MAX_SIZE], odd[MAX_SIZE];
8     int i, size, even_count = 0, odd_count = 0;
9
10    /* Input size of the array */
11    printf("Enter size of the array: ");
12    scanf("%d", &size);
13
14    /* Input elements in array */
15    printf("Enter elements in the array: ");
16    for(i=0; i<size; i++)
17    {
18        scanf("%d", &arr[i]);
19    }
20
21    /* Traverse array and store even elements in even[] and odd elements in odd[] */
22    for(i=0; i<size; i++)
23    {
24        if(arr[i] % 2 == 0)
25        {
26            even[even_count] = arr[i];
27            even_count++;
28        }
29        else
30        {
31            odd[odd_count] = arr[i];
32            odd_count++;
33        }
34    }
35
36    /* Print even elements */
37    printf("\nEven elements: ");
38    for(i=0; i<even_count; i++)
39    {
40        printf("%d ", even[i]);
41    }
42
43    /* Print odd elements */
44    printf("\nOdd elements: ");
45    for(i=0; i<odd_count; i++)
46    {
47        printf("%d ", odd[i]);
48    }
49
50    return 0;
51 }
```

C:\Users\91961\Documents\6. x + v - □ ×

Enter elements in the array: 3
4

Even elements: 4
Odd elements: 3

Process exited after 27.11 seconds with return value 0
Press any key to continue . . . |

7.maximum and minimum.cpp

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int n, i;
6      printf("Enter no. of elements in an array: ");
7      scanf("%d", &n);
8
9      int arr[n];
10     printf("Enter the elements: ");
11     for (i = 0; i < n; i++) {
12         scanf("%d", &arr[i]);
13     }
14
15     // initialize max and min with the first element of the array
16     int max = arr[0], min = arr[0];
17
18     // Loop through the array to find the maximum and minimum values
19     for (i = 1; i < n; i++) {
20         if (arr[i] > max) {
21             max = arr[i];
22         }
23         if (arr[i] < min) {
24             min = arr[i];
25         }
26     }
27
28     printf("Maximum value: %d\n", max);
29     printf("Minimum value: %d\n", min);
30
31     return 0;
}
```

```
C:\Users\91961\Documents\7. × + ▾
Enter no. of elements in an array: 5
Enter the elements: 1 2 3 4 5
Maximum value: 5
Minimum value: 1

-----
Process exited after 48.34 seconds with return value 0
Press any key to continue . . . |
```

8.size ass 2.cpp

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int arr[100], freq[100];
6     int size, i, j, count;
7
8     printf("Enter the size of the array: ");
9     scanf("%d", &size);
10
11     printf("Enter the elements of the array:\n");
12     for (i = 0; i < size; i++) {
13         scanf("%d", &arr[i]);
14         freq[i] = -1;
15     }
16
17     for (i = 0; i < size; i++) {
18         count = 1;
19         for (j = i+1; j < size; j++) {
20             if (arr[i] == arr[j]) {
21                 count++;
22                 freq[j] = 0;
23             }
24             if (freq[i] != 0) {
25                 freq[i] = count;
26             }
27         }
28     }
29
30     printf("Frequency of all elements of array:\n");
31     for (i = 0; i < size; i++) {
32         if (freq[i] != 0) {
33             printf("%d occurs %d times\n", arr[i], freq[i]);
34         }
35     }
36
37     return 0;
}
```

Close tab (Ctrl+F4)

```
C:\Users\91961\Documents\8.
Enter the size of the array: 3
Enter the elements of the array:
8
7
9
Frequency of all elements of array:
8 occurs 1 times
7 occurs 1 times
9 occurs 1 times

-----
Process exited after 23.13 seconds with return value 0
Press any key to continue . . .
```


9.sort array assi 2.cpp

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int arr[100], n, i, j, temp;
6
7     printf("Enter the value of N: ");
8     scanf("%d", &n);
9
10    printf("Enter the numbers:\n");
11    for(i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14
15    // sorting the array in descending order
16    for(i = 0; i < n; i++) {
17        for(j = i+1; j < n; j++) {
18            if(arr[i] < arr[j]) {
19                temp = arr[i];
20                arr[i] = arr[j];
21                arr[j] = temp;
22            }
23        }
24    }
25
26    printf("The numbers arranged in descending order are given below:\n");
27    for(i = 0; i < n; i++) {
28        printf("%d\n", arr[i]);
29    }
30
31    return 0;
}
```

C:\Users\91961\Documents\9.

Enter the value of N: 5

Enter the numbers:

234

780

130

56

90

The numbers arranged in descending order are given below:

780

234

130

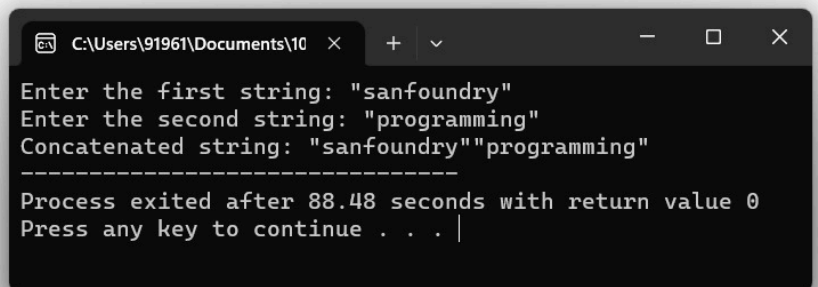
90

56

Process exited after 69.73 seconds with return value 0
Press any key to continue . . .

10. concatenating two strings.cpp

```
1  #include <stdio.h>
2  #include <string.h>
3
4  int main() {
5      char str1[100], str2[100];
6
7      printf("Enter the first string: ");
8      fgets(str1, 100, stdin);
9      str1[strcspn(str1, "\n")] = 0; // remove newline character
10
11     printf("Enter the second string: ");
12     fgets(str2, 100, stdin);
13     str2[strcspn(str2, "\n")] = 0; // remove newline character
14
15     strcat(str1, str2);
16     printf("Concatenated string: %s", str1);
17
18     return 0;
19 }
```



```
C:\Users\91961\Documents\10
Enter the first string: "sanfoundry"
Enter the second string: "programming"
Concatenated string: "sanfoundry" "programming"
-----
Process exited after 88.48 seconds with return value 0
Press any key to continue . . . |
```