

# Assignment Code for Programming in C

Name: Mahdi Hasan Shuvo

 Roll No: 251-115-030

 Subject: Programming in C

 Department: CSE-62

 Submission Date: 31-07-2025

## Index for C Programming Assignment

1. Reverse Array Order
2. Sum of Array Elements
3. Count Duplicate Elements
4. Print Unique Elements
5. Find Max & Min Elements
6. Separate Odd & Even Numbers
7. Reverse String Characters
8. Count Words in String
9. Max Frequency Character in String
10. Swap Two Numbers
11. Check Prime Number
12. Find Largest Array Element
13. Max Frequency Character in String (Same as 9)

## 1. Reverse the Order of Elements in an Array

Code:

The screenshot shows a code editor with the following C code:

```
1 #include <stdio.h>
2
3 int main() {
4     int n;
5
6     printf("Input the number of elements to store in the array: ");
7     scanf("%d", &n);
8
9     int arr[n];
10    printf("Input %d number of elements in the array:\n", n);
11    for(int i = 0; i < n; i++) {
12        printf("element - %d : ", i);
13        scanf("%d", &arr[i]);
14    }
15
16    printf("\nThe values stored into the array are: ");
17    for(int i = 0; i < n; i++) {
18        printf("%d ", arr[i]);
19    }
20
21    printf("\nThe values stored into the array in reverse are: ");
22    for(int i = n - 1; i >= 0; i--) {
23        printf("%d ", arr[i]);
24    }
25
26    return 0;
27 }
```

The output window shows the execution results:

```
Input the number of elements to store in the array: 2
Input 2 number of elements in the array:
element - 0 : 12
element - 1 : 2

The values stored into the array are: 12 2
The values stored into the array in reverse are: 2 12

--- Code Execution Successful ---
```

## 2. Sum of All Elements of an Array

Code:

The screenshot shows a code editor with the following C code:

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n, sum = 0;
5
6     printf("Input the number of elements to be stored in the array: ");
7     scanf("%d", &n);
8
9     int arr[n];
10    printf("Input %d elements in the array:\n", n);
11    for(int i = 0; i < n; i++) {
12        printf("element - %d : ", i);
13        scanf("%d", &arr[i]);
14        sum += arr[i];
15    }
16
17    printf("Sum of all elements stored in the array is: %d\n", sum);
18    return 0;
19 }
```

The output window shows the execution results:

```
Input the number of elements to be stored in the array: 2
Input 2 elements in the array:
element - 0 : 8
element - 1 : 9

Sum of all elements stored in the array is: 17

--- Code Execution Successful ---
```

## 3. Count Total Number of Duplicate Elements in an Array

Code:

The screenshot shows a code editor window with a dark theme. The left pane contains the C code for finding duplicates in an array. The right pane shows the output of the code execution.

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n, count = 0;
5
6     printf("Input the number of elements to be stored in the array: ");
7     scanf("%d", &n);
8
9     int arr[n];
10    printf("Input %d elements in the array:\n", n);
11    for(int i = 0; i < n; i++) {
12        printf("element - %d : ", i);
13        scanf("%d", &arr[i]);
14    }
15
16    for(int i = 0; i < n; i++) {
17        for(int j = i + 1; j < n; j++) {
18            if(arr[i] == arr[j]) {
19                count++;
20                break;
21            }
22        }
23    }
24
25    printf("Total number of duplicate elements found in the array is: %d\n", count);
26    return 0;
27 }
```

Output:

```
Input the number of elements to be stored in the array: 4
Input 4 elements in the array:
element - 0 : 2
element - 1 : 3
element - 2 : 4
element - 3 : 2
Total number of duplicate elements found in the array is: 1
--- Code Execution Successful ---
```

## 4. Print All Unique Elements in an Array

Code:

The screenshot shows a code editor window with a dark theme. The left pane contains the C code for printing unique elements in an array. The right pane shows the output of the code execution.

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Input the number of elements to be stored in the array: ");
6     scanf("%d", &n);
7     int arr[n];
8     printf("Input %d elements in the array:\n", n);
9     for(int i = 0; i < n; i++) {
10        printf("element - %d : ", i);
11        scanf("%d", &arr[i]);
12    }
13
14    printf("The unique elements found in the array are: ");
15    for(int i = 0; i < n; i++) {
16        int is_unique = 1;
17        for(int j = 0; j < n; j++) {
18            if(i != j && arr[i] == arr[j]) {
19                is_unique = 0;
20                break;
21            }
22        }
23        if(is_unique) {
24            printf("%d ", arr[i]);
25        }
26    }
27    return 0;
28 }
```

Output:

```
Input the number of elements to be stored in the array: 4
Input 4 elements in the array:
element - 0 : 3
element - 1 : 4
element - 2 : 5
element - 3 : 6
The unique elements found in the array are: 3 4 5 6
--- Code Execution Successful ---
```

## 5. Find the Maximum and Minimum Elements in an Array

Code:

```

main.c | Run | Output
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Input the number of elements to be stored in the array: ");
6     scanf("%d", &n);
7     int arr[n];
8     printf("Input %d elements in the array:\n", n);
9     for(int i = 0; i < n; i++) {
10         printf("element - %d : ", i);
11         scanf("%d", &arr[i]);
12     }
13     int max = arr[0], min = arr[0];
14     for(int i = 1; i < n; i++) {
15         if(arr[i] > max) {
16             max = arr[i];
17         }
18         if(arr[i] < min) {
19             min = arr[i];
20         }
21     }
22     printf("The maximum element is: %d\n", max);
23     printf("The minimum element is: %d\n", min);
24     return 0;
25 }

```

Input the number of elements to be stored in the array: 4  
Input 4 elements in the array:  
element - 0 : 5  
element - 1 : 6  
element - 2 : 3  
element - 3 : 2  
The maximum element is: 6  
The minimum element is: 2  
\*\*\* Code Execution Successful \*\*\*

## 6. Separate Odd and Even Integers into Separate Arrays

Code:

```

main.c | Run | Output
1 #include <stdio.h>
2
3 int main() {
4     int n;
5     printf("Input the number of elements to be stored in the array: ");
6     scanf("%d", &n);
7     int arr[n], even[n], odd[n], even_count = 0, odd_count = 0;
8     printf("Input %d elements in the array:\n", n);
9     for(int i = 0; i < n; i++) {
10         printf("element - %d : ", i);
11         scanf("%d", &arr[i]);
12     }
13     for(int i = 0; i < n; i++) {
14         if(arr[i] % 2 == 0) {
15             even[even_count++] = arr[i];
16         } else {
17             odd[odd_count++] = arr[i];
18         }
19     }
20     printf("The Even elements are: ");
21     for(int i = 0; i < even_count; i++) {
22         printf("%d ", even[i]);
23     }
24     printf("\nThe Odd elements are: ");
25     for(int i = 0; i < odd_count; i++) {
26         printf("%d ", odd[i]);
27     }
28     return 0;
29 }

```

Input the number of elements to be stored in the array: 4  
Input 4 elements in the array:  
element - 0 : 3  
element - 1 : 5  
element - 2 : 6  
element - 3 : 7  
The Even elements are: 6  
The Odd elements are: 3 5 7  
\*\*\* Code Execution Successful \*\*\*

## 7. Print Individual Characters of a String in Reverse Order

Code:

```
C main.c > ...
2  /*__DEVOLPER__ = " MAHDI HASAN SHUVO ";
4  //__LinkedIn__ = "mahdi-hasan-shuvo";*/
5  #include <stdio.h>
6  #include <string.h>
Windsurf: Refactor | Explain | Generate Function Comment | X | ⌂
7  int main() {
8      char str[100];
9      printf("Input the string: ");
10     fgets(str, sizeof(str), stdin);
11     int len = strlen(str);
12     printf("The characters in reverse order are: ");
13     for(int i = len - 2; i >= 0; i--) { // len - 2 to exclude newline character
14         printf("%c ", str[i]);
15     }
16
17     return 0;
18 }
19
```

PROBLEMS    OUTPUT    DEBUG CONSOLE    **TERMINAL**    PORTS

```
PS M:\Gast project\Clint-58> & 'm:\Gast project\Clint-58\main.exe'
Input the string: Mahdi hasan shuvo
The characters in reverse order are: o v u h s   n a s a h   i d h a M
PS M:\Gast project\Clint-58> [ ]
```

## 8. Count the Total Number of Words in a String

Code:

C main.c > main()

```

5  #include <stdio.h>
6  #include <string.h>
7
8  int main() {
9      char str[100];
10     printf("Input the string: ");
11     fgets(str, sizeof(str), stdin);
12     int count = 1; // For the first word
13     for(int i = 0; str[i] != '\0'; i++) {
14         if(str[i] == ' ' || str[i] == '\n') {
15             count++;
16         }
17     }
18     printf("Total number of words in the string: %d\n", count);
19     return 0;
20 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS M:\Gast project\Clint-58> & 'm:\Gast project\Clint-58\main.exe'
   
Input the string: Mahdi hasan Shuvo
   
Total number of words in the string: 4
- PS M:\Gast project\Clint-58> [ ]

## 9. Find the Maximum Number of Characters in a String (Frequency)

Code:

C Online Compiler

main.c		Output
<pre> 1 #include &lt;stdio.h&gt; 2 #include &lt;string.h&gt; 3 4 int main() { 5     char str[100]; 6     int freq[256] = {0}; // Frequency array for all ASCII characters 7 8     printf("Input the string: "); 9     fgets(str, sizeof(str), stdin); 10 11    for(int i = 0; str[i] != '\0'; i++) { 12        freq[str[i]]++; 13    } 14 15    int max_freq = 0; 16    char max_char; 17    for(int i = 0; i &lt; 256; i++) { 18        if(freq[i] &gt; max_freq) { 19            max_freq = freq[i]; 20            max_char = i; 21        } 22    } 23 24    printf("The Highest frequency of character '%c' appears %d times\n", max_char, max_freq); 25    return 0; 26 }</pre>	Input the string: Mahdi shubo The Highest frequency of character 'h' appears 2 times === Code Execution Successful ===	

## 10. Swap Two Numbers Using a Function

Code:

The screenshot shows a code editor interface with a dark theme. On the left, the code file 'main.c' is open, displaying the following C code:

```
1 #include <stdio.h>
2
3 void swap(int *a, int *b) {
4     int temp = *a;
5     *a = *b;
6     *b = temp;
7 }
8
9 int main() {
10     int num1, num2;
11
12     printf("Input 1st number: ");
13     scanf("%d", &num1);
14     printf("Input 2nd number: ");
15     scanf("%d", &num2);
16
17     printf("Before swapping: n1 = %d, n2 = %d\n", num1, num2);
18
19     swap(&num1, &num2);
20
21     printf("After swapping: n1 = %d, n2 = %d\n", num1, num2);
22
23 }
```

On the right, there is an 'Output' panel showing the results of running the code. The output is:

```
Input 1st number: 8
Input 2nd number: 3
Before swapping: n1 = 8, n2 = 3
After swapping: n1 = 3, n2 = 8
*** Code Execution Successful ***
```

## 11. Check Whether a Number is Prime or Not Using a Function

Code:

The screenshot shows a code editor interface with a dark theme. On the left, the code file 'main.c' is open, displaying the following C code:

```
1 #include <stdio.h>
2
3 int is_prime(int num) {
4     if(num <= 1) return 0;
5     for(int i = 2; i * i <= num; i++) {
6         if(num % i == 0) return 0;
7     }
8     return 1;
9 }
10
11 int main() {
12     int num;
13
14     printf("Input a positive number: ");
15     scanf("%d", &num);
16
17     if(is_prime(num)) {
18         printf("The number %d is a prime number.\n", num);
19     } else {
20         printf("The number %d is not a prime number.\n", num);
21     }
22
23 }
```

On the right, there is an 'Output' panel showing the results of running the code. The output is:

```
Input a positive number: 22
The number 22 is not a prime number.
*** Code Execution Successful ***
```

## 12. Find the Largest Element of an Array Using a Function

## Code:

The screenshot shows a code editor interface with a dark theme. On the left is the code editor pane containing the file `main.c`. The code defines a function `largest` that finds the maximum value in an array. It also contains a `main` function that reads the size of the array and its elements from standard input, then prints the maximum value to standard output. The code is annotated with line numbers from 1 to 28. On the right is the output pane, which displays the execution results. It shows the user inputting 2 elements (3 and 2), the program printing their values, and then outputting the largest element (3) and a success message.

```
main.c | Run | Output
1 #include <stdio.h>
2
3 int largest(int arr[], int n) {
4     int mex = arr[0];
5     for(int i = 1; i < n; i++) {
6         if(arr[i] > mex) {
7             mex = arr[i];
8         }
9     }
10    return mex;
11 }
12
13 int main() {
14     int n;
15     printf("Input the number of elements to be stored in the array: ");
16     scanf("%d", &n);
17
18     int arr[n];
19     printf("Input %d elements in the array:\n", n);
20     for(int i = 0; i < n; i++) {
21         printf("element - %d : ", i);
22         scanf("%d", &arr[i]);
23     }
24
25     printf("The largest element in the array is: %d\n", largest(arr, n));
26     return 0;
27 }
```

```
Input the number of elements to be stored in the array: 2
Input 2 elements in the array:
element - 0 : 3
element - 1 : 2
The largest element in the array is: 3
--- Code Execution Successful ---
```