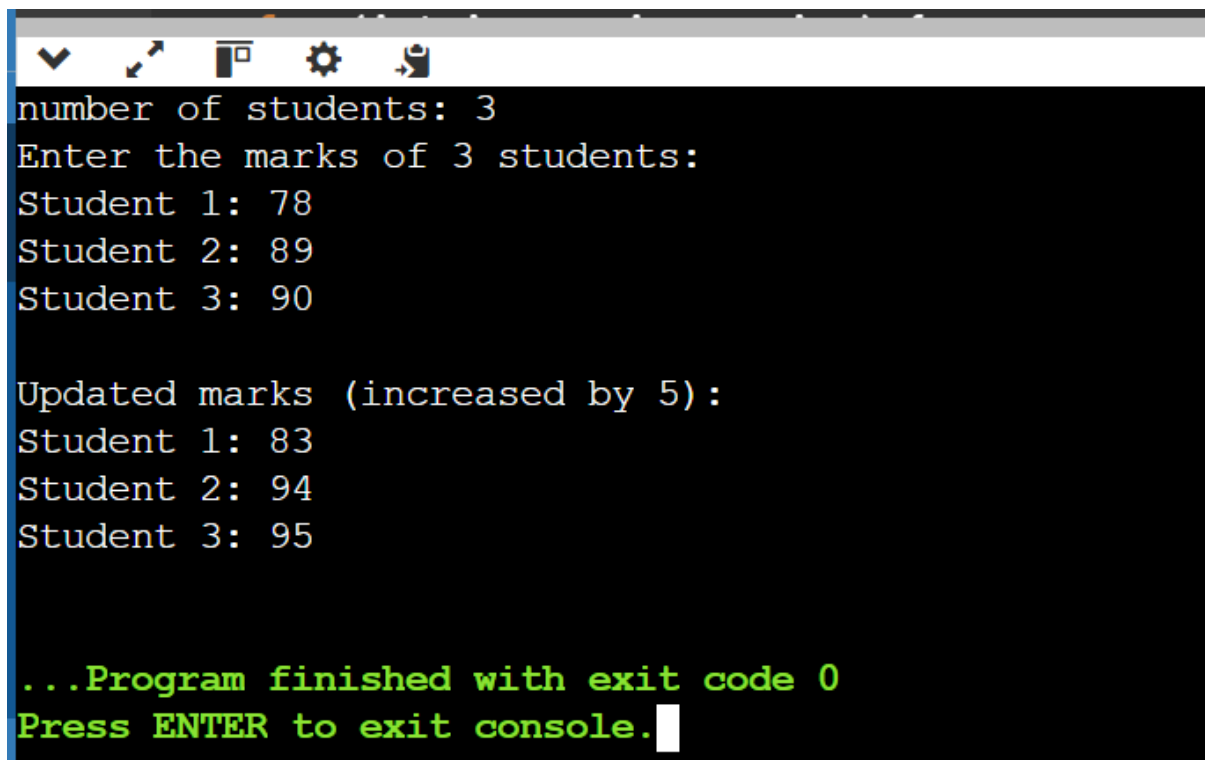


# ASSIGNMENT 2

## QUES 1

```
int main() {  
    int n;  
    printf(" number of students: ");  
    scanf("%d", &n);  
    int marks[n];  
    printf("Enter the marks of %d students:\n", n);  
    for (int i = 0; i < n; i++) {  
        printf("Student %d: ", i + 1);  
        scanf("%d", &marks[i]);  
    }  
    for (int i = 0; i < n; i++) {  
        marks[i] += 5;  
    }  
    printf("\nUpdated marks (increased by 5):\n");  
    for (int i = 0; i < n; i++) {  
        printf("Student %d: %d\n", i + 1, marks[i]);  
    }  
}
```



```
number of students: 3
Enter the marks of 3 students:
Student 1: 78
Student 2: 89
Student 3: 90

Updated marks (increased by 5):
Student 1: 83
Student 2: 94
Student 3: 95

...Program finished with exit code 0
Press ENTER to exit console.
```

QUES 2

```
#include <stdio.h>
```

```
int main() {
    int n;
    printf(" number of students: ");
    scanf("%d", &n);
    int marks[n];
    printf("Enter the marks of %d students:\n", n);
    for (int i = 0; i < n; i++) {
        printf("Student %d: ", i + 1);
        scanf("%d", &marks[i]);
    }
    printf("\nGrades of students:\n");
    for (int i = 0; i < n; i++) {
```

```

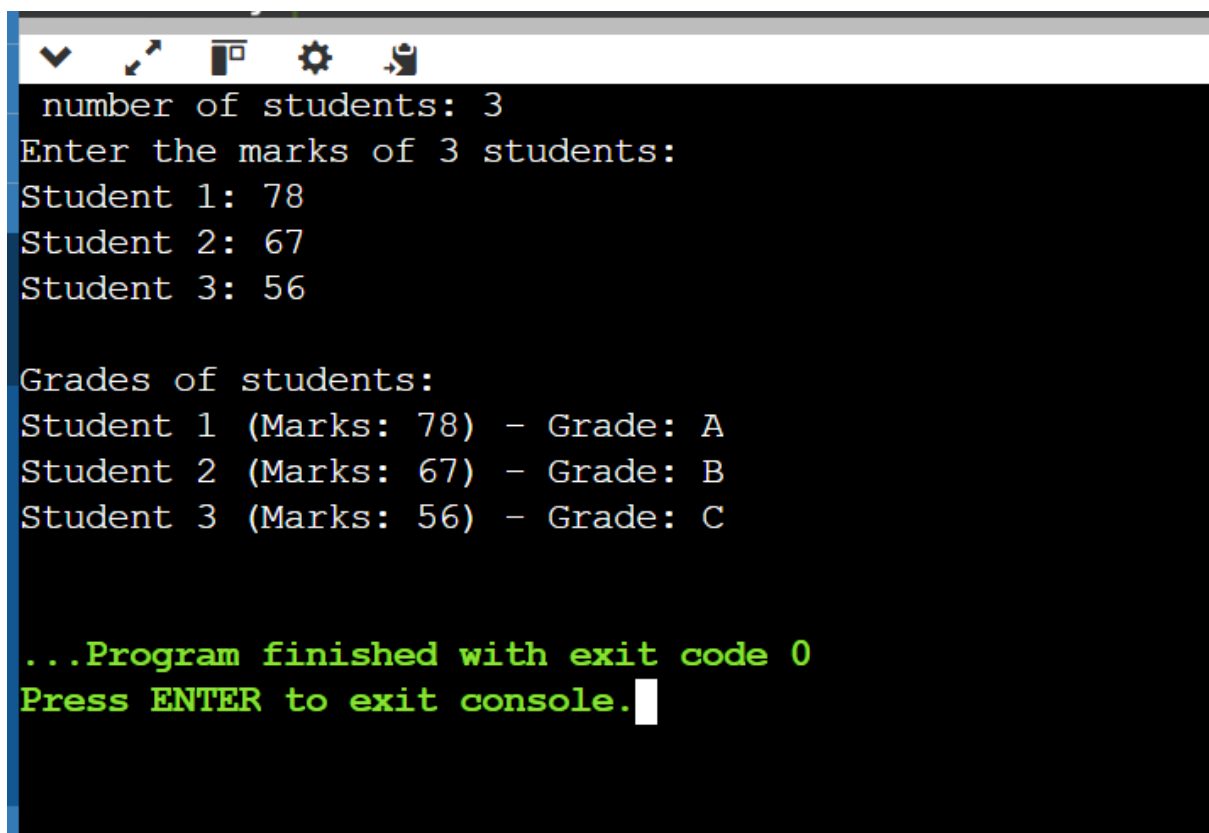
char grade;

if (marks[i] >= 75) {
    grade = 'A';
} else if (marks[i] >= 60) {
    grade = 'B';
} else if (marks[i] >= 40) {
    grade = 'C';
} else {
    grade = 'D';
}

printf("Student %d (Marks: %d) - Grade: %c\n", i + 1, marks[i], grade);
}

return 0;
}

```



```

number of students: 3
Enter the marks of 3 students:
Student 1: 78
Student 2: 67
Student 3: 56

Grades of students:
Student 1 (Marks: 78) - Grade: A
Student 2 (Marks: 67) - Grade: B
Student 3 (Marks: 56) - Grade: C

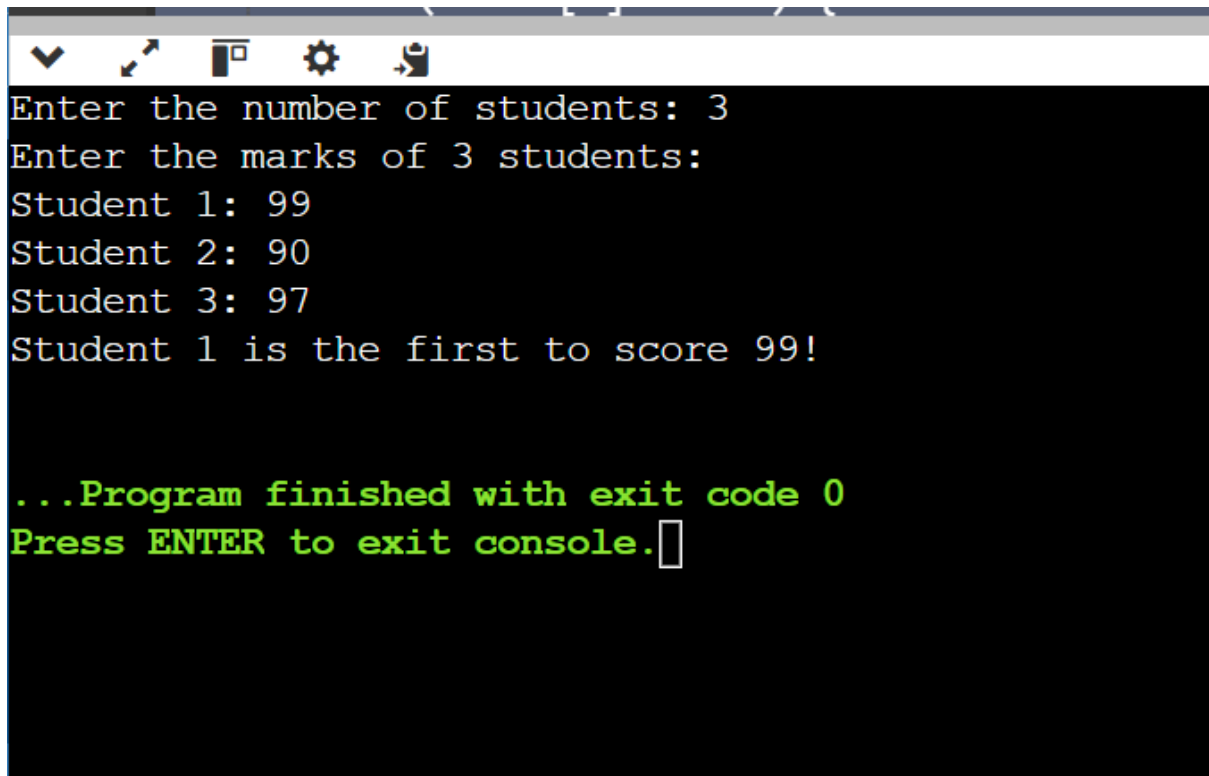
...Program finished with exit code 0
Press ENTER to exit console.

```

QUES 3

```
#include <stdio.h>
```

```
int main() {  
    int n;  
    printf("Enter the number of students: ");  
    scanf("%d", &n);  
    int marks[n];  
    printf("Enter the marks of %d students:\n", n);  
    for (int i = 0; i < n; i++) {  
        printf("Student %d: ", i + 1);  
        scanf("%d", &marks[i]);  
    }  
    int found = 0;  
    for (int i = 0; i < n; i++) {  
        if (marks[i] == 99) {  
            printf("Student %d is the first to score 99!\n", i + 1);  
            found = 1;  
            break;  
        }  
    }  
    if (!found) {  
        printf("No student scored 99.\n");  
    }  
  
    return 0;  
}
```



```
Enter the number of students: 3
Enter the marks of 3 students:
Student 1: 99
Student 2: 90
Student 3: 97
Student 1 is the first to score 99!

...Program finished with exit code 0
Press ENTER to exit console.
```

QUES 4

```
#include <stdio.h>
```

```
int main() {
    int n;
    printf("Enter the number of students: ");
    scanf("%d", &n);
    int marks[n];
    printf("Enter the marks of %d students:\n", n);
    for (int i = 0; i < n; i++) {
        printf("Student %d: ", i + 1);
        scanf("%d", &marks[i]);
    }
    int count = 0;
    printf("\nStudents who scored 99:\n");
    for (int i = 0; i < n; i++) {
```

```

    if (marks[i] == 99) {
        printf("Student %d scored 99\n", i + 1);
        count++;
    }
}

if (count > 0) {
    printf("\nTotal number of students who scored 99: %d\n", count);
} else {
    printf("\nNo students scored 99.\n");
}

return 0;
}

```

```

Enter the number of students: 3
Enter the marks of 3 students:
Student 1: 78
Student 2: 99
Student 3: 89

Students who scored 99:
Student 2 scored 99

Total number of students who scored 99: 1

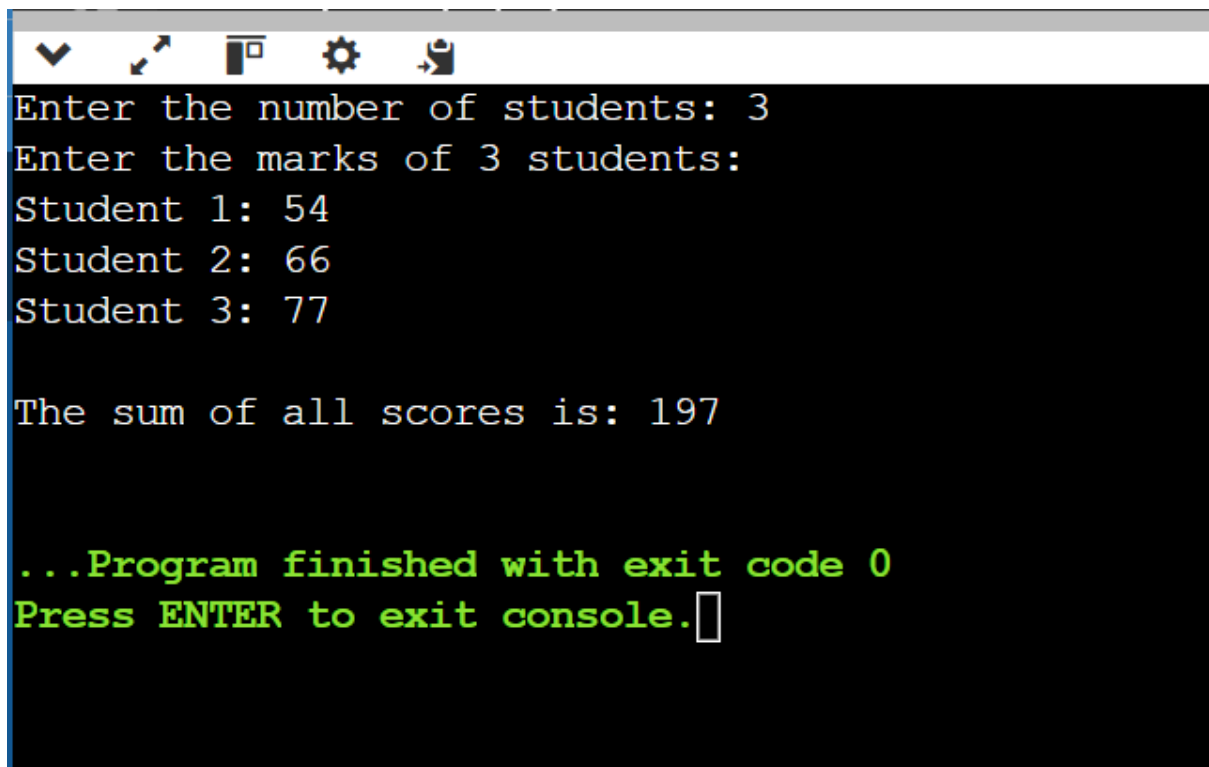
...Program finished with exit code 0
Press ENTER to exit console.

```

QUES 5

```
#include <stdio.h>
```

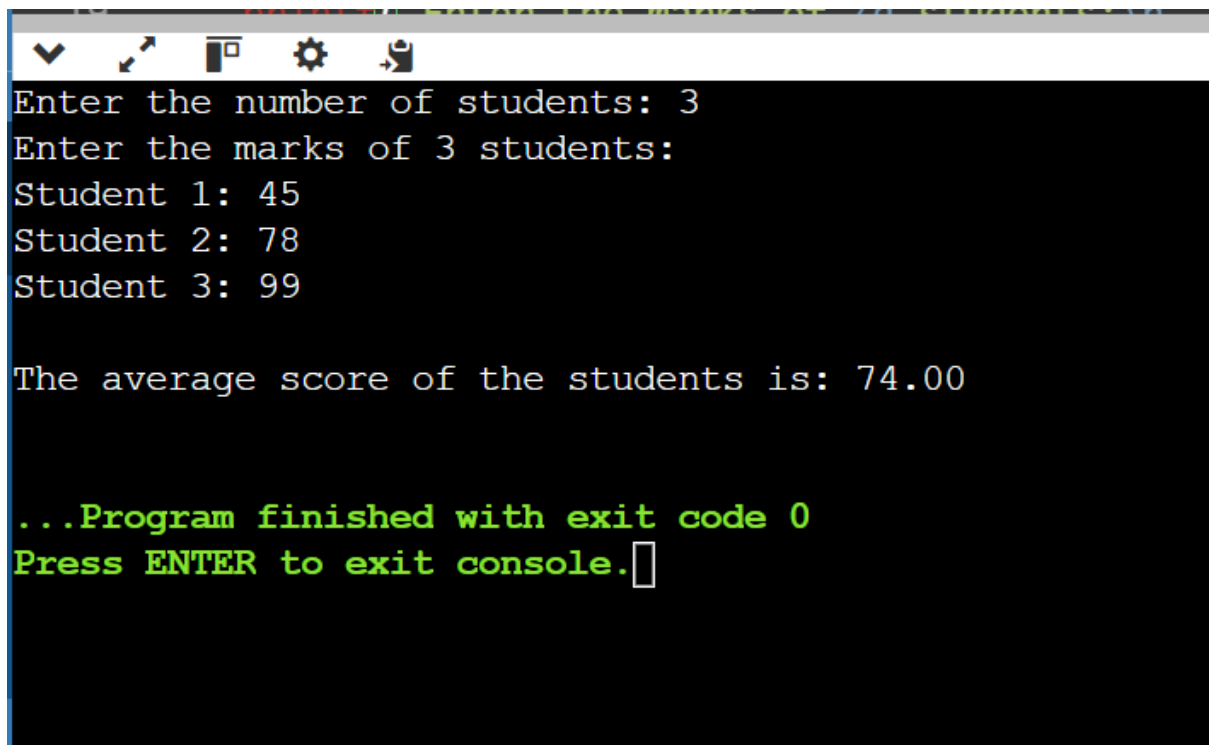
```
int main() {  
    int n;  
    printf("Enter the number of students: ");  
    scanf("%d", &n);  
    int marks[n];  
    int sum = 0;  
    printf("Enter the marks of %d students:\n", n);  
    for (int i = 0; i < n; i++) {  
        printf("Student %d: ", i + 1);  
        scanf("%d", &marks[i]);  
    }  
    for (int i = 0; i < n; i++) {  
        sum += marks[i];  
    }  
    printf("\nThe sum of all scores is: %d\n", sum);  
    return 0;  
}
```

A screenshot of a console window showing the execution of a C program. The window has a title bar with standard icons. The output text is as follows:  
Enter the number of students: 3  
Enter the marks of 3 students:  
Student 1: 54  
Student 2: 66  
Student 3: 77  
  
The sum of all scores is: 197  
  
...Program finished with exit code 0  
Press ENTER to exit console.  
The program logic, shown in the previous block, involves asking for the number of students, then for each student's mark, and finally calculating and displaying the sum of all marks.

```
#include <stdio.h>
```

```
int main() {  
    int n;  
    printf("Enter the number of students: ");  
    scanf("%d", &n);  
    int marks[n];  
    int sum = 0;  
    printf("Enter the marks of %d students:\n", n);  
    for (int i = 0; i < n; i++) {  
        printf("Student %d: ", i + 1);  
        scanf("%d", &marks[i]);  
    }  
    for (int i = 0; i < n; i++) {  
        sum += marks[i];  
    }  
    float average = (float)sum / n;  
    printf("\nThe average score of the students is: %.2f\n", average);  
    return 0;  
}
```





```
Enter the number of students: 3
Enter the marks of 3 students:
Student 1: 45
Student 2: 78
Student 3: 99

The average score of the students is: 74.00

...Program finished with exit code 0
Press ENTER to exit console.
```

QUES 7

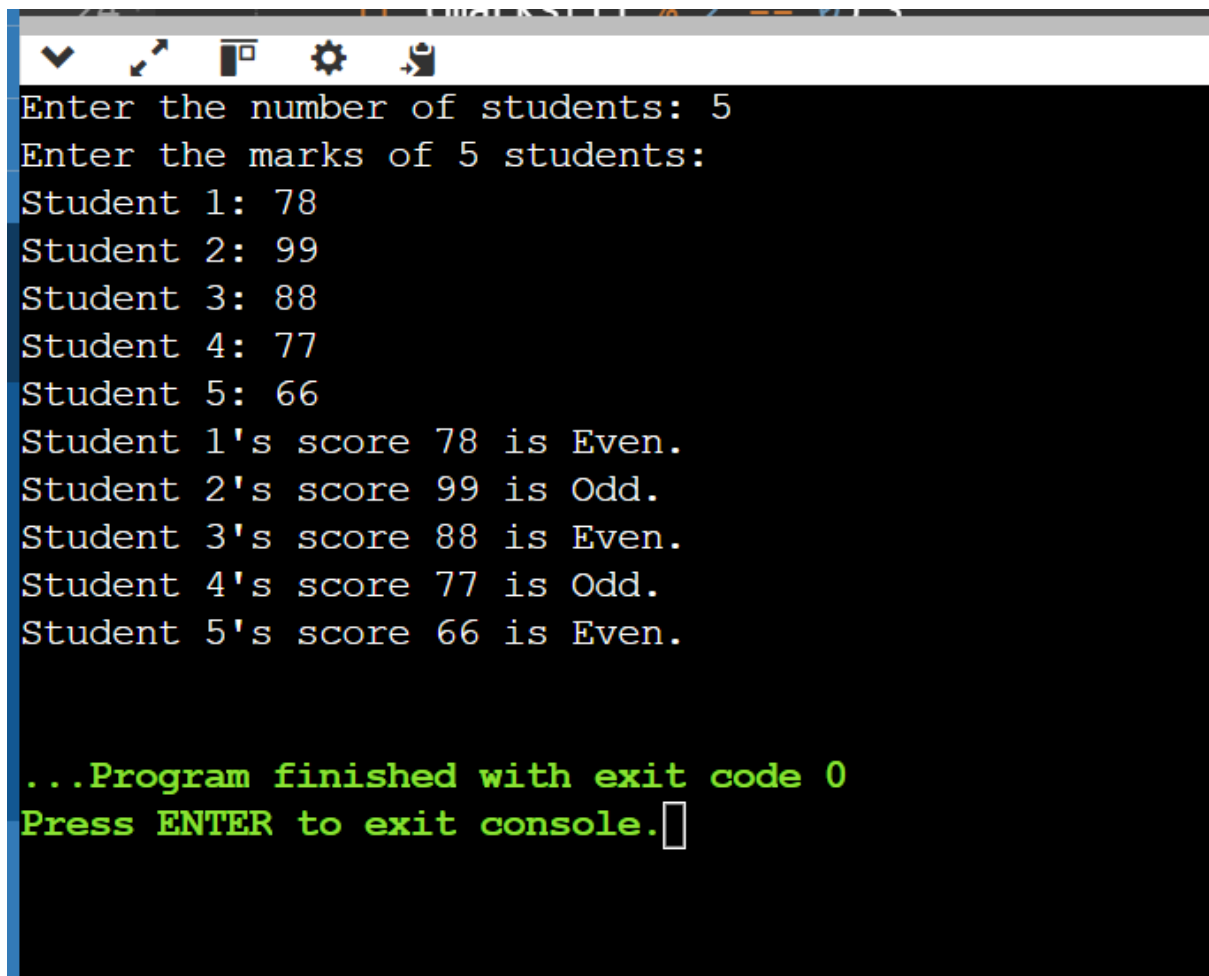
```
#include <stdio.h>
```

```
int main() {
    int n;
    printf("Enter the number of students: ");
    scanf("%d", &n);
    int marks[n];
    printf("Enter the marks of %d students:\n", n);
    for (int i = 0; i < n; i++) {
        printf("Student %d: ", i + 1);
        scanf("%d", &marks[i]);
    }
    for (int i = 0; i < n; i++) {
        if (marks[i] % 2 == 0) {
            printf("Student %d's score %d is Even.\n", i + 1, marks[i]);
        }
    }
}
```

```

    } else {
        printf("Student %d's score %d is Odd.\n", i + 1, marks[i]);
    }
}
return 0;
}

```



```

Enter the number of students: 5
Enter the marks of 5 students:
Student 1: 78
Student 2: 99
Student 3: 88
Student 4: 77
Student 5: 66
Student 1's score 78 is Even.
Student 2's score 99 is Odd.
Student 3's score 88 is Even.
Student 4's score 77 is Odd.
Student 5's score 66 is Even.

...Program finished with exit code 0
Press ENTER to exit console.

```

QUES 8

```
#include <stdio.h>
```

```

int main() {
    int n;

    printf("Enter the number of students: ");

```

```
scanf("%d", &n);

int marks[n];

printf("Enter the marks of %d students:\n", n);

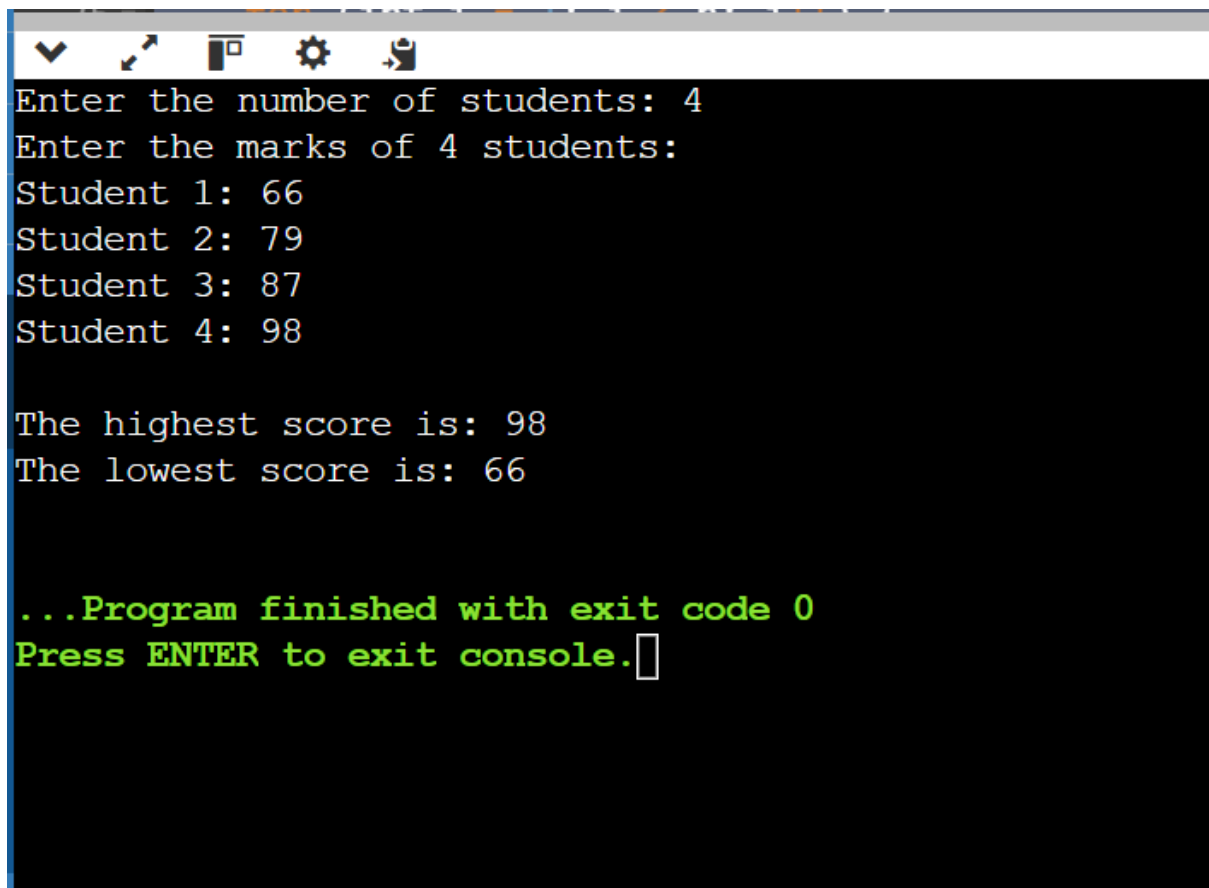
for (int i = 0; i < n; i++) {
    printf("Student %d: ", i + 1);
    scanf("%d", &marks[i]);
}

int max = marks[0];
int min = marks[0];

for (int i = 1; i < n; i++) {
    if (marks[i] > max) {
        max = marks[i];
    }
    if (marks[i] < min) {
        min = marks[i];
    }
}

printf("\nThe highest score is: %d\n", max);
printf("The lowest score is: %d\n", min);

return 0;
}
```



```
Enter the number of students: 4
Enter the marks of 4 students:
Student 1: 66
Student 2: 79
Student 3: 87
Student 4: 98

The highest score is: 98
The lowest score is: 66

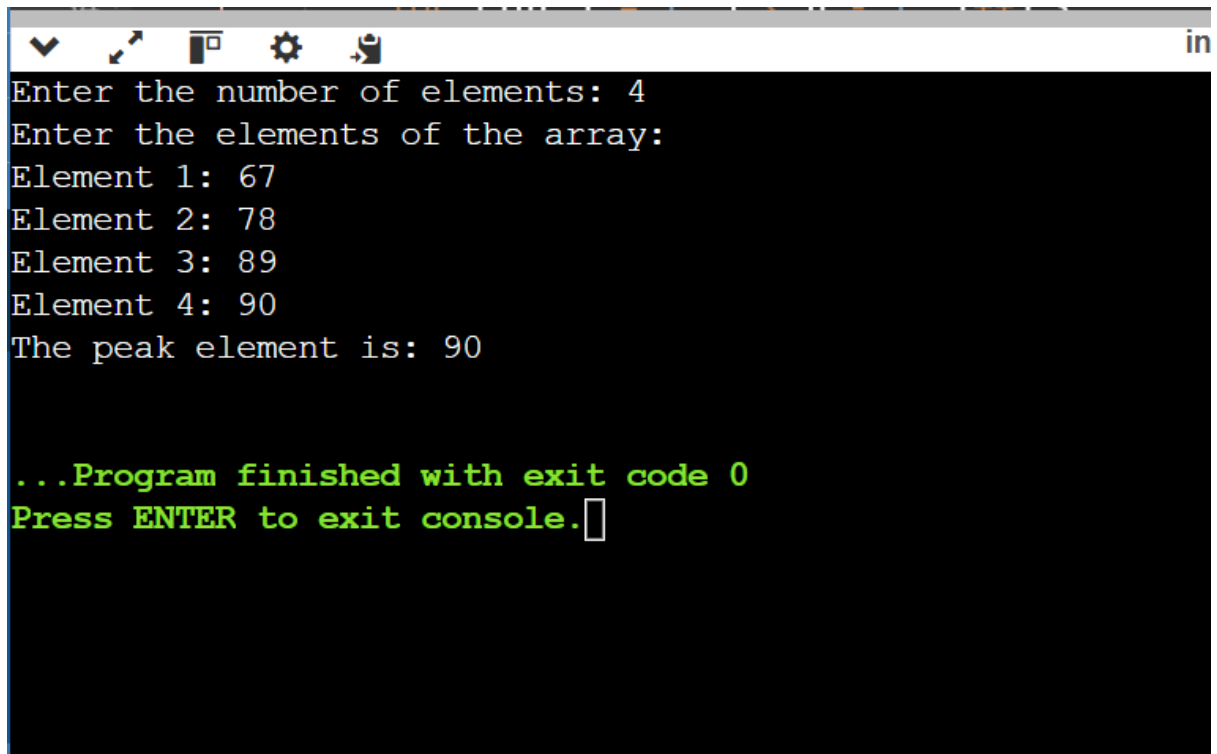
...Program finished with exit code 0
Press ENTER to exit console.
```

QUES 9

```
#include <stdio.h>
```

```
int main() {
    int n;
    printf("Enter the number of elements: ");
    scanf("%d", &n);
    int arr[n];
    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++) {
        printf("Element %d: ", i + 1);
        scanf("%d", &arr[i]);
    }
}
```

```
if (n == 1) {  
    printf("The peak element is: %d\n", arr[0]);  
} else {  
    if (arr[0] >= arr[1]) {  
        printf("The peak element is: %d\n", arr[0]);  
    }  
    else if (arr[n - 1] >= arr[n - 2]) {  
        printf("The peak element is: %d\n", arr[n - 1]);  
    }  
    else {  
        for (int i = 1; i < n - 1; i++) {  
            if (arr[i] >= arr[i - 1] && arr[i] >= arr[i + 1]) {  
                printf("The peak element is: %d\n", arr[i]);  
                break;  
            }  
        }  
    }  
}  
return 0;  
}
```



```
Enter the number of elements: 4
Enter the elements of the array:
Element 1: 67
Element 2: 78
Element 3: 89
Element 4: 90
The peak element is: 90

...Program finished with exit code 0
Press ENTER to exit console.
```

QUES 10

```
#include <stdio.h>

#include <math.h>

int is_prime(int num) {
    if (num <= 1) {
        return 0;
    }
    for (int i = 2; i <= sqrt(num); i++) {
        if (num % i == 0) {
            return 0;
        }
    }
    return 1;
}

int main() {
    int n;

    printf("Enter the number of elements: ");
    scanf("%d", &n);
```

```
int arr[n];

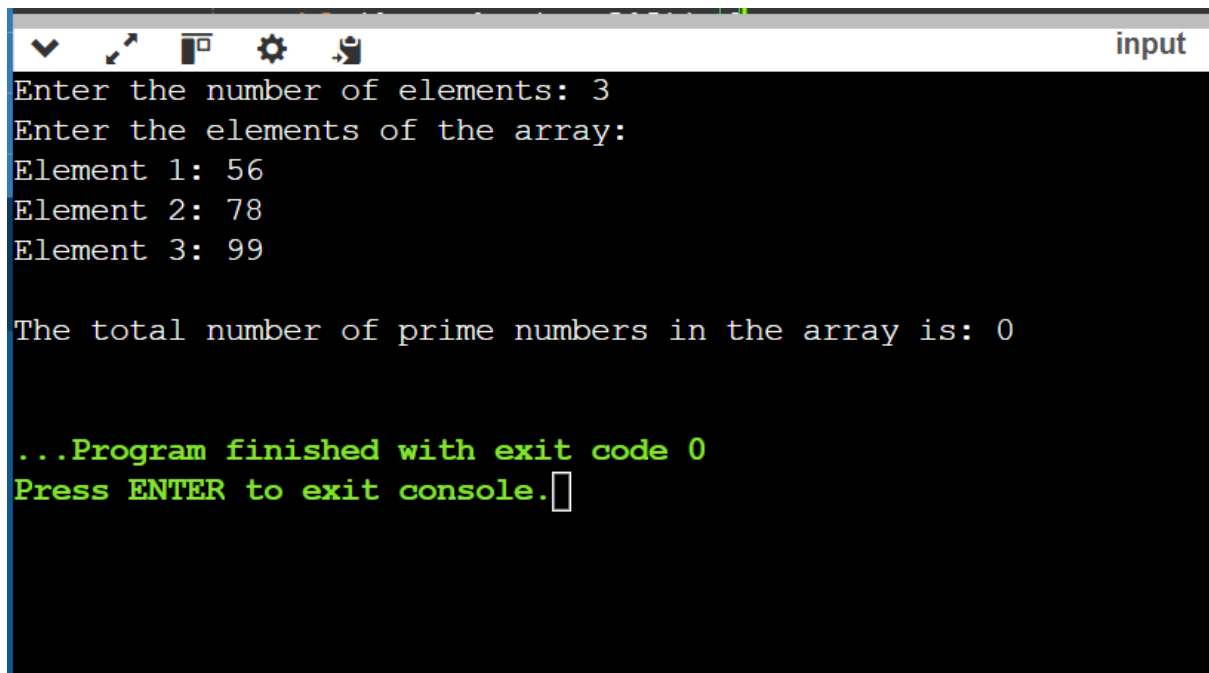
printf("Enter the elements of the array:\n");

for (int i = 0; i < n; i++) {
    printf("Element %d: ", i + 1);
    scanf("%d", &arr[i]);
}

int prime_count = 0;
for (int i = 0; i < n; i++) {
    if (is_prime(arr[i])) {
        prime_count++;
    }
}

printf("\nThe total number of prime numbers in the array is: %d\n", prime_count);

return 0;
}
```



```
input
Enter the number of elements: 3
Enter the elements of the array:
Element 1: 56
Element 2: 78
Element 3: 99

The total number of prime numbers in the array is: 0

...Program finished with exit code 0
Press ENTER to exit console.
```

## QUES 11

```
#include <stdio.h>
```

```
void printArray(int arr[], int size) {
```

```
    for (int i = 0; i < size; i++) {
```

```
        printf("%d ", arr[i]);
```

```
    }
```

```
    printf("\n");
```

```
}
```

```
void insertAtFront(int arr[], int *size, int element) {
```

```
    for (int i = *size; i > 0; i--) {
```

```
        arr[i] = arr[i - 1];
```

```
    }
```

```
    arr[0] = element;
```

```
    (*size)++;
```

```
}
```

```
void insertAtPosition(int arr[], int *size, int element, int position) {
```

```
    if (position < 0 || position > *size) {
```

```
        printf("Invalid position!\n");
```

```
        return;
```

```
    }
```

```
    for (int i = *size; i > position; i--) {
```

```
        arr[i] = arr[i - 1];
```

```
    }
```

```
    arr[position] = element;
```

```
    (*size)++;
```

```
}
```

```
void insertAtEnd(int arr[], int *size, int element) {
```

```
    arr[*size] = element;
```

```
    (*size)++;
```

```
}
```

```
int main() {
```



```
int arr[100]; // array with a maximum size of 100
```

```
int size, choice, element, position;
```

```
printf("Enter the initial size of the array: ");
```

```
scanf("%d", &size);
```

```
printf("Enter %d elements of the array:\n", size);
```

```
for (int i = 0; i < size; i++) {
```

```
    scanf("%d", &arr[i]);
```

```
}
```

```
printf("Array before insertion:\n");
```

```
printArray(arr, size);
```

```
printf("\nChoose an insertion option:\n");
```

```
printf("1. Insert at the front\n");
```

```
printf("2. Insert at a specific position\n");
```

```
printf("3. Insert at the end\n");
```

```
printf("Enter your choice: ");
```

```
scanf("%d", &choice);
```

```
printf("Enter the element to insert: ");
```

```
scanf("%d", &element);
```

```
switch (choice) {
```

```
    case 1:
```

```
        insertAtFront(arr, &size, element);
```

```
        break;
```

```
    case 2:
```

```
        printf("Enter the position (0 to %d): ", size);
```

```
        scanf("%d", &position);
```

```
        insertAtPosition(arr, &size, element, position);
```

```
        break;
    case 3:
        insertAtEnd(arr, &size, element);
        break;
    default:
        printf("Invalid choice!\n");
        return 1;
}

printf("Array after insertion:\n");
printArray(arr, size);


return 0;
}
```

```
Enter the initial size of the array: 4
Enter 4 elements of the array:
5
6
7
7
Array before insertion:
5 6 7 7

Choose an insertion option:
1. Insert at the front
2. Insert at a specific position
3. Insert at the end
Enter your choice: 3
Enter the element to insert: 6
Array after insertion:
5 6 7 7 6

...Program finished with exit code 0
Press ENTER to exit console.
```

QUES 12

```
#include <stdio.h>
```

```
void printArray(int arr[], int size) {
```

```
    for (int i = 0; i < size; i++) {
```

```
        printf("%d ", arr[i]);
```

```
    }
```

```
    printf("\n");
```

```
}
```

```
void deleteFront(int arr[], int *size) {
```

```
    if (*size == 0) {
```

```
        printf("Array is empty. Cannot delete front element.\n");
```

```

        return;
    }
    for (int i = 0; i < *size - 1; i++) {
        arr[i] = arr[i + 1];
    }
    (*size)--;
}

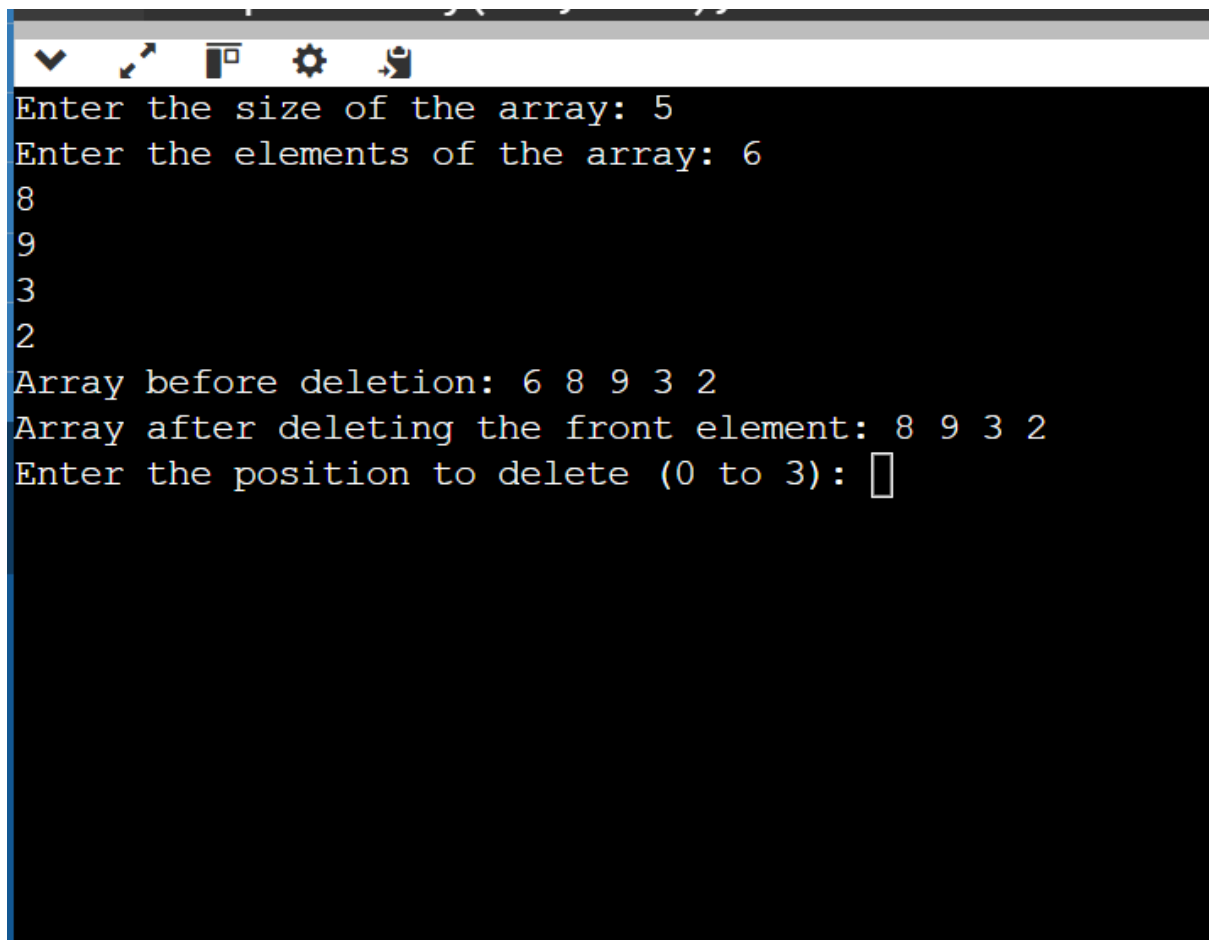
void deleteAtPosition(int arr[], int *size, int position) {
    if (position < 0 || position >= *size) {
        printf("Invalid position. Please provide a valid position.\n");
        return;
    }
    for (int i = position; i < *size - 1; i++) {
        arr[i] = arr[i + 1];
    }
    (*size)--;
}

void deleteEnd(int *size) {
    if (*size == 0) {
        printf("Array is empty. Cannot delete end element.\n");
        return;
    }
    (*size)--;
}

int main() {
    int arr[100], size, position;
    printf("Enter the size of the array: ");
    scanf("%d", &size);
    printf("Enter the elements of the array: ");
    for (int i = 0; i < size; i++) {
        scanf("%d", &arr[i]);
    }
}

```

```
}  
  
printf("Array before deletion: ");  
printArray(arr, size);  
deleteFront(arr, &size);  
printf("Array after deleting the front element: ");  
printArray(arr, size);  
printf("Enter the position to delete (0 to %d): ", size - 1);  
scanf("%d", &position);  
deleteAtPosition(arr, &size, position);  
printf("Array after deleting element at position %d: ", position);  
printArray(arr, size);  
deleteEnd(&size);  
printf("Array after deleting the last element: ");  
printArray(arr, size);  
  
return 0;  
}
```

A screenshot of a C++ IDE window. The window has a title bar with standard icons (checkmark, zoom, window, settings, save). The main area is a black console with white text. The text shows a sequence of prompts and user inputs for an array deletion program. The prompts are: 'Enter the size of the array:', 'Enter the elements of the array:', 'Array before deletion:', 'Array after deleting the front element:', and 'Enter the position to delete (0 to 3):'. The user inputs are: '5', '6', '8', '9', '3', '2', '8 9 3 2', and an empty box for the position.

```
Enter the size of the array: 5
Enter the elements of the array: 6
8
9
3
2
Array before deletion: 6 8 9 3 2
Array after deleting the front element: 8 9 3 2
Enter the position to delete (0 to 3): 
```

QUES 13

```
#include <stdio.h>
```

```
void rotateClockwiseByOne(int arr[], int n) {
```

```
    int last = arr[n - 1];
```

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

```
        arr[i] = arr[i - 1];
```

```
    }
```

```
    arr[0] = last;
```

```
}
```

```
void printArray(int arr[], int n) {
```

```
    for (int i = 0; i < n; i++) {
```

```
        printf("%d ", arr[i]);
```

```
    }
```

```
    printf("\n");
```

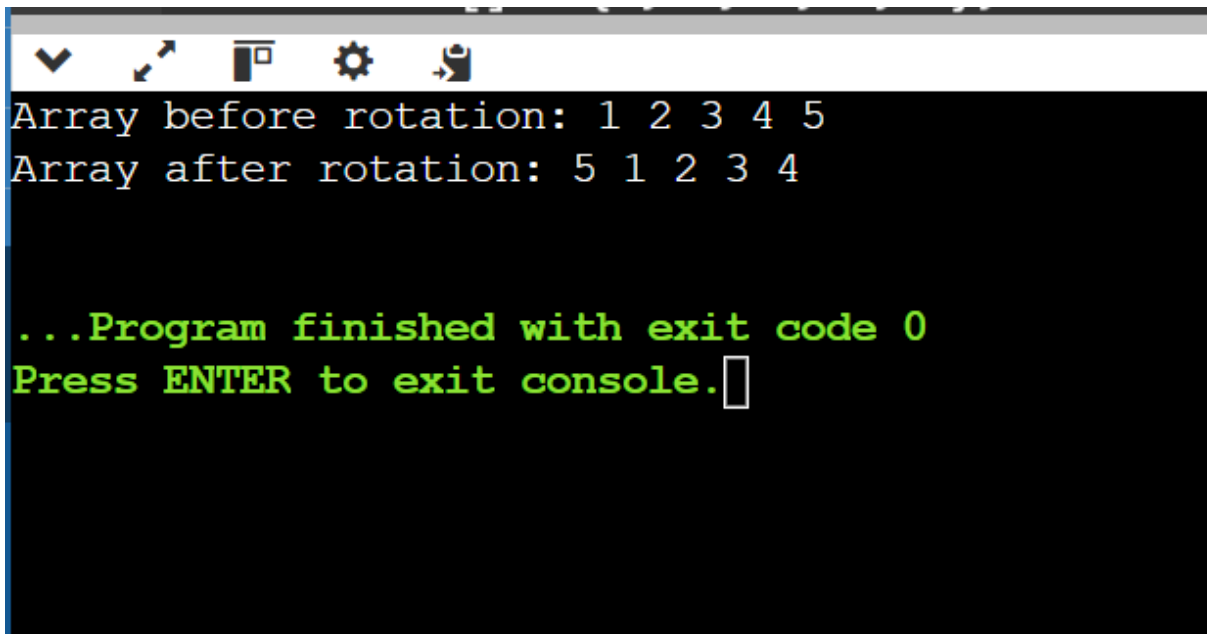
```
}
```

```

int main() {
    int arr[] = {1, 2, 3, 4, 5};
    int n = sizeof(arr) / sizeof(arr[0]);
    printf("Array before rotation: ");
    printArray(arr, n);
    rotateClockwiseByOne(arr, n);
    printf("Array after rotation: ");
    printArray(arr, n);

    return 0;
}

```



```

Array before rotation: 1 2 3 4 5
Array after rotation: 5 1 2 3 4

...Program finished with exit code 0
Press ENTER to exit console.

```

#### QUES 14

```

#include <stdio.h>

void printDuplicates(int arr[], int n) {
    int freq[1000] = {0};
    int foundDuplicate = 0;
    for (int i = 0; i < n; i++) {
        freq[arr[i]]++;
    }
}

```

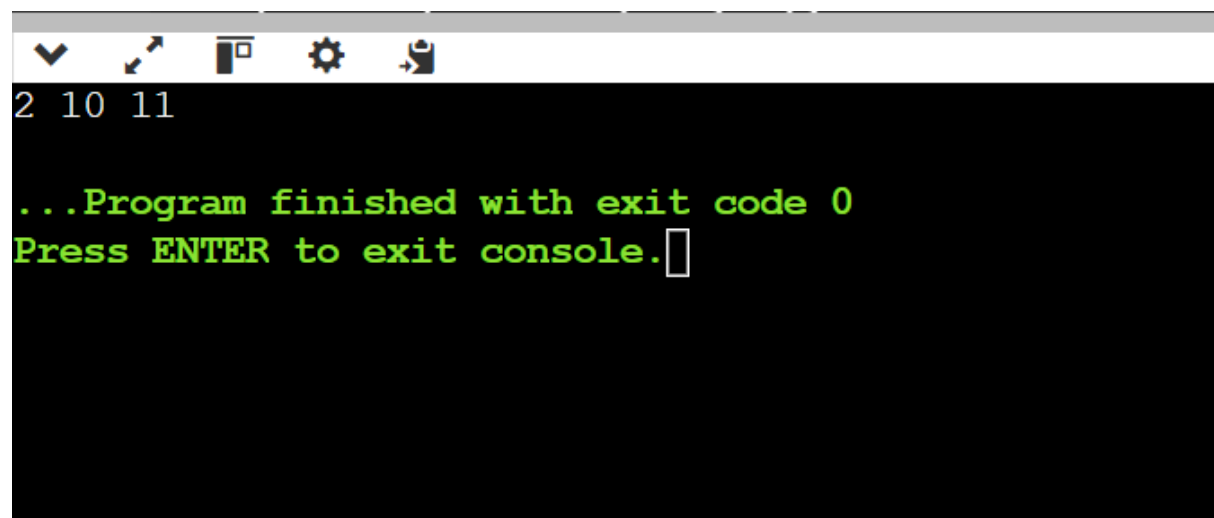
```

    }
    for (int i = 0; i < n; i++) {
        if (freq[arr[i]] > 1) {
            printf("%d ", arr[i]);
            freq[arr[i]] = 0;
            foundDuplicate = 1;
        }
    }
    if (!foundDuplicate) {
        printf("-1\n");
    }
}

int main() {
    int arr[] = {2, 10, 10, 100, 2, 10, 11, 2, 11, 2};
    int n = sizeof(arr) / sizeof(arr[0]);
    printDuplicates(arr, n);

    return 0;
}

```



```

cmd
2 10 11

...Program finished with exit code 0
Press ENTER to exit console.

```