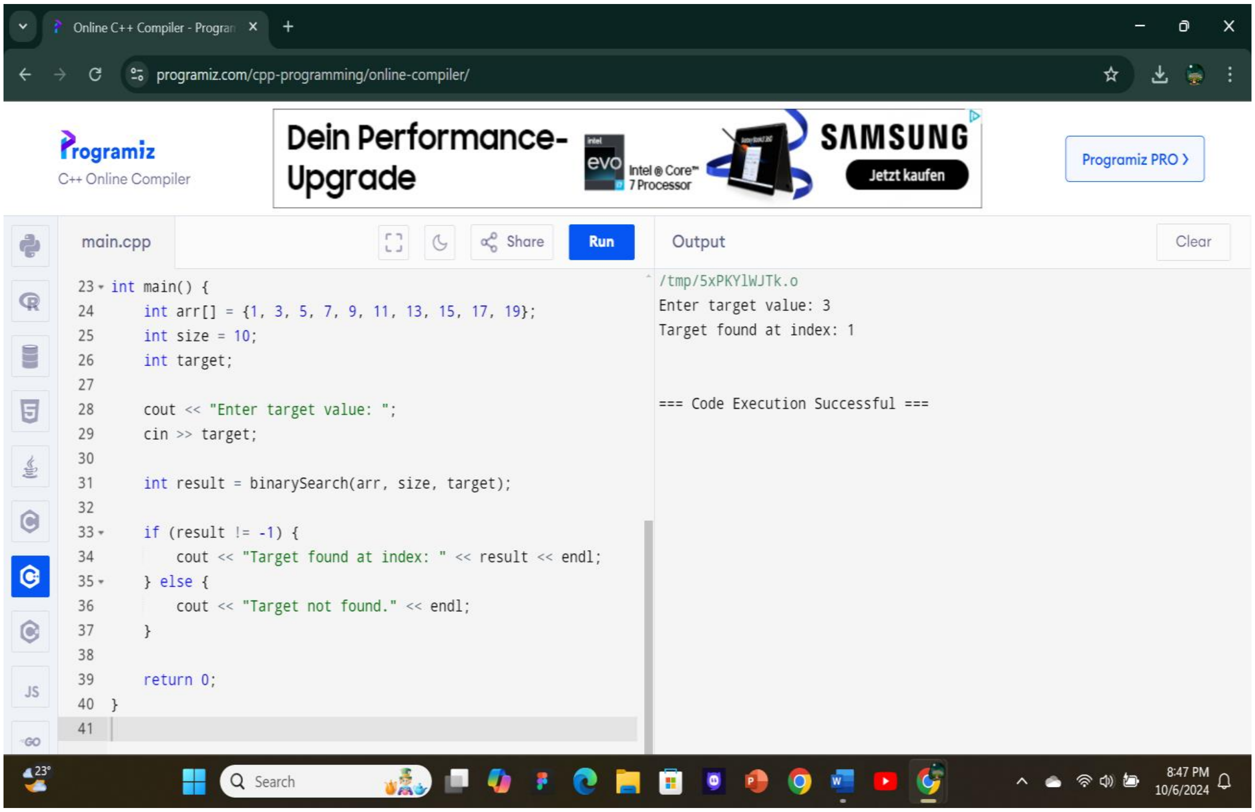
Name:- Haider Ali

SAP ID:- 53109

SUBJECT:-DSA

```
//*****Question#1*****
#include <iostream>
using namespace std;

int binarySearch(int arr[], int size, int target) {
    int left = 0;
    int right = size - 1;

    while (left <= right) {
        int mid = (left + right) / 2;

        if (arr[mid] == target) {
            return mid;
        } else if (arr[mid] < target) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
    }

    return -1;
}

int main() {
    int arr[] = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19};
    int size = 10;
    int target;

    cout << "Enter target value: ";
    cin >> target;

    int result = binarySearch(arr, size, target);

    if (result != -1) {
        cout << "Target found at index: " << result << endl;
    } else {
        cout << "Target not found." << endl;
    }

    return 0;
}
```

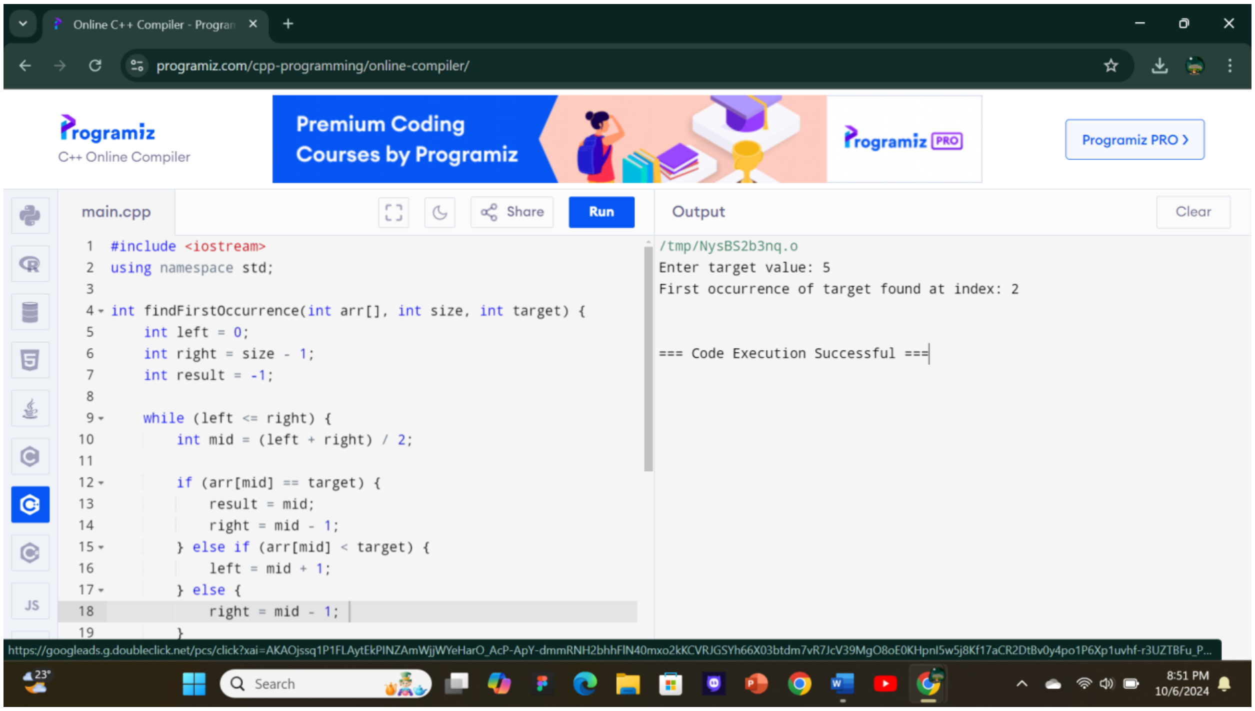


```cpp
int main() {
    int arr[] = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19};
    int size = 10;
    int target;

    cout << "Enter target value: ";
    cin >> target;

    int result = binarySearch(arr, size, target);

    if (result != -1) {
        cout << "Target found at index: " << result << endl;
    } else {
        cout << "Target not found." << endl;
    }

    return 0;
}
```

```
/tmp/5xPKYlWJTk.o
Enter target value: 3
Target found at index: 1


=== Code Execution Successful ===
```

```cpp
//*****Question#2*****
#include <iostream>
using namespace std;

int findFirstOccurrence(int arr[], int size, int target) {
   int left = 0;
   int right = size - 1;
   int result = -1;

   while (left <= right) {
      int mid = (left + right) / 2;

      if (arr[mid] == target) {
            result = mid;
            right = mid - 1;
        } else if (arr[mid] < target) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
    }

    return result; // return index or -1 if not found
}

int main() {
    int arr[] = {1, 3, 5, 7, 7, 7, 9, 11, 13, 15};
    int size = 10;
    int target;

    cout << "Enter target value: ";
    cin >> target;

    int result = findFirstOccurrence(arr, size, target);

    if (result != -1) {
        cout << "First occurrence of target found at index: " << result << endl;
    } else {
        cout << "Target not found." << endl;
    }

    return 0;
```

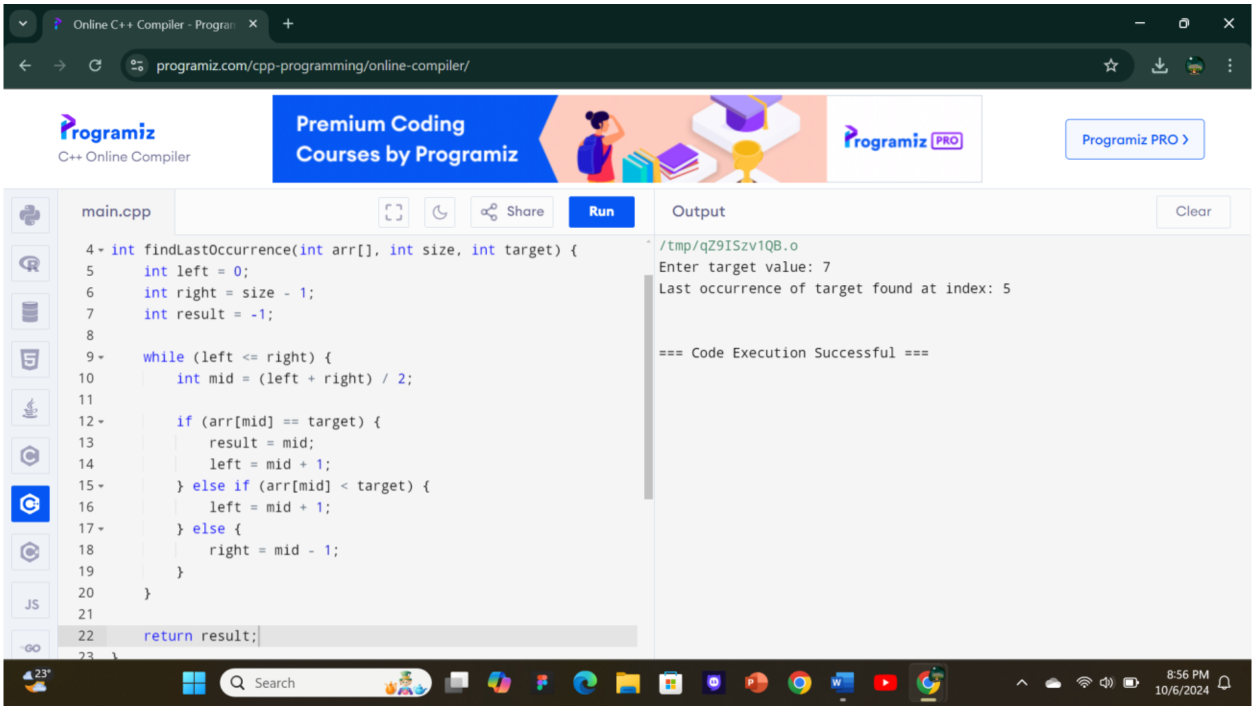
```
}
```


Online C++ Compiler - Progran
programiz.com/cpp-programming/online-compiler/
Programiz
C++ Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
Programiz PRO ›
main.cpp
Share
Run
Output
Clear
Search
8:51 PM
10/6/2024

```
#include <iostream>
using namespace std;

int findFirstOccurrence(int arr[], int size, int target) {
    int left = 0;
    int right = size - 1;
    int result = -1;

    while (left <= right) {
        int mid = (left + right) / 2;

        if (arr[mid] == target) {
            result = mid;
            right = mid - 1;
        } else if (arr[mid] < target) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
```

```
/tmp/NysBS2b3nq.o
Enter target value: 5
First occurrence of target found at index: 2


=== Code Execution Successful ===
```

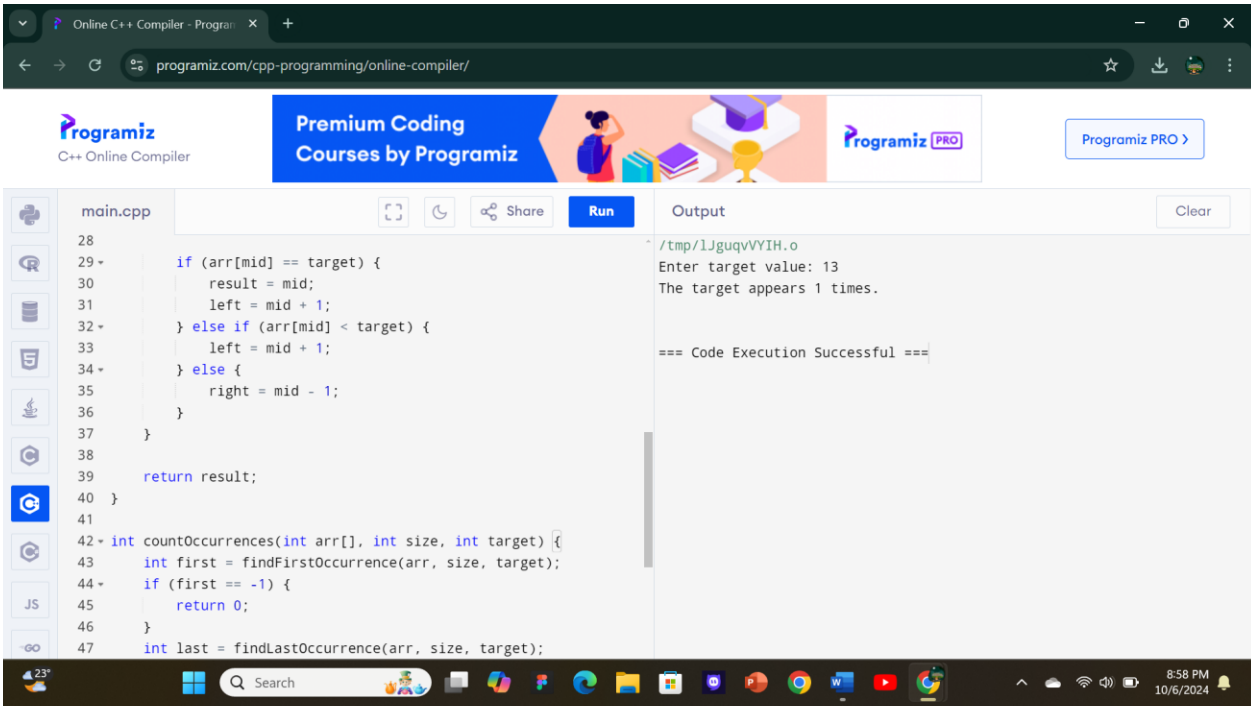
```
//*****Question#3*****
#include <iostream>
using namespace std;

int findLastOccurrence(int arr[], int size, int target) {
    int left = 0;
    int right = size - 1;
    int result = -1;

    while (left <= right) {
        int mid = (left + right) / 2; // simpler mid calculation

        if (arr[mid] == target) {
            result = mid; // store index
            left = mid + 1; // keep searching the right side
        } else if (arr[mid] < target) {
            left = mid + 1; // search right half
        } else {
            right = mid - 1; // search left half
        }
    }

    return result; // return index or -1 if not found
}

int main() {
    int arr[] = {1, 3, 5, 7, 7, 7, 9, 11, 13, 15};
    int size = 10; // directly setting size for simplicity
    int target;

    cout << "Enter target value: ";
    cin >> target;

    int result = findLastOccurrence(arr, size, target);

    if (result != -1) {
        cout << "Last occurrence of target found at index: " << result << endl;
    } else {
        cout << "Target not found." << endl;
    }

    return 0;
}
```



```
int findLastOccurrence(int arr[], int size, int target) {
    int left = 0;
    int right = size - 1;
    int result = -1;

    while (left <= right) {
        int mid = (left + right) / 2;

        if (arr[mid] == target) {
            result = mid;
            left = mid + 1;
        } else if (arr[mid] < target) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
    }

    return result;
}
```

```
/tmp/qZ9ISzv1QB.o
Enter target value: 7
Last occurrence of target found at index: 5


=== Code Execution Successful ===
```

```
//*****Question#4*****
#include <iostream>
using namespace std;


int findFirstOccurrence(int arr[], int size, int target) {
    int left = 0, right = size - 1, result = -1;


    while (left <= right) {
        int mid = (left + right) / 2; // simpler mid calculation


        if (arr[mid] == target) {
            result = mid;
            right = mid - 1; // search left half for the first occurrence
        } else if (arr[mid] < target) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
    }

    return result;
}

int findLastOccurrence(int arr[], int size, int target) {
    int left = 0, right = size - 1, result = -1;

    while (left <= right) {
        int mid = (left + right) / 2; // simpler mid calculation

        if (arr[mid] == target) {
            result = mid;
            left = mid + 1;
        } else if (arr[mid] < target) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
    }

    return result;
}

int countOccurrences(int arr[], int size, int target) {
    int first = findFirstOccurrence(arr, size, target);
    if (first == -1) {
        return 0;
    }
    int last = findLastOccurrence(arr, size, target);
    return last - first + 1; // number of occurrences
}

int main() {
    int arr[] = {1, 3, 5, 7, 7, 7, 9, 11, 13, 15};
    int size = 10;
    int target;

    cout << "Enter target value: ";
    cin >> target;

    int count = countOccurrences(arr, size, target);

    if (count > 0) {
        cout << "The target appears " << count << " times." << endl;
    } else {
        cout << "Target not found." << endl;
    }

    return 0;
}
```


Online C++ Compiler - Program
programiz.com/cpp-programming/online-compiler/
Programiz
C++ Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
Programiz PRO ›
main.cpp
Share
Run
Output
Clear
Search
8:58 PM
10/6/2024

```cpp

        if (arr[mid] == target) {
            result = mid;
            left = mid + 1;
        } else if (arr[mid] < target) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
    }

    return result;
}

int countOccurrences(int arr[], int size, int target) {
    int first = findFirstOccurrence(arr, size, target);
    if (first == -1) {
        return 0;
    }
    int last = findLastOccurrence(arr, size, target);
```

```
/tmp/lJguqvVYIH.o
Enter target value: 13
The target appears 1 times.


=== Code Execution Successful ===
```