Assignment 1

```
#include<iostream>
using namespace std;
// Recursive binary search function
int binarySearchRecursive(int arr[], int left, int right, int target) {
if (left <= right) {
int mid = left + (right - left) / 2;
if (arr[mid] == target)
return mid;
if (arr[mid] > target)
return binarySearchRecursive(arr, left, mid - 1, target);
return binarySearchRecursive(arr, mid + 1, right, target);
}
return -1;
}
// Non-recursive binary search function
int binarySearchIterative(int arr[], int n, int target) {
int left = 0, right = n - 1;
while (left <= right) {
int mid = left + (right - left) / 2;
if (arr[mid] == target)
return mid;
if (arr[mid] > target)
right = mid - 1;
else
left = mid + 1;
}
return -1;
}
int main() {
int n, target, choice;
```

```
char choice1;
cout << "Enter the number of elements in the array: ";
cin >> n;
int arr[n];
cout << "Enter sorted elements in the array: ";
for (int i = 0; i < n; i++) {
cin >> arr[i];
}
cout << "Enter the element to search: ";
cin >> target;
do {
cout << "Choose the search method:\n";
cout << "1. Recursive Binary Search\n";
cout << "2. Non-Recursive Binary Search\n";
cout << "Enter your choice (1 or 2): ";
cin >> choice;
int result;
switch (choice) {
case 1:
result = binarySearchRecursive(arr, 0, n - 1, target);
break;
case 2:
result = binarySearchIterative(arr, n, target);
break;
default:
cout << "Invalid choice" << endl;
return 1;
}
if (result != -1)
cout << "Element is present at index: " << result << endl;
else
```

```
cout << "Element is not present in the array" << endl;

cout << "Do you want to continue? (y/n): ";

cin >> choice1;

} while (choice1 == 'y' || choice1 == 'Y');

return 0;

}
```

Output: