**Name Nadeem ullah**

**Sap id 56031**

**Semester CS3-2**

**Question 1**

#include <iostream>

using namespace std;

void printActiveArray(int arr[], int low, int high) {

cout << "Current active items: [";

for (int i = low; i <= high; ++i) {

cout << arr[i];

if (i != high) cout << ", ";

}

cout << "]" << endl;

}

int binarySearch(int arr[], int size, int target) {

int low = 0, high = size - 1;

while (low <= high) {

printActiveArray(arr, low, high);

int mid = low + (high - low) / 2;

if (arr[mid] == target) {

return mid;

} else if (arr[mid] > target) {

high = mid - 1;

} else {

low = mid + 1;

}

}

return -1;

}

int main() {

int arr[] = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19};

int size = sizeof(arr) / sizeof(arr[0]);

int target;

cout << "Enter the target value to search for: ";

cin >> target;

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

if (result != -1) {

cout << "Target found at index: " << result << endl;

} else {

cout << "Target not found in the array." << endl;

}

return 0;

}

**Question 2**

#include <iostream>

using namespace std;

void printActiveArray(int arr[], int low, int high) {

cout << "Current active items: [";

for (int i = low; i <= high; ++i) {

cout << arr[i];

if (i != high) cout << ", ";

}

cout << "]" << endl;

}

int findFirstOccurrence(int arr[], int size, int target) {

int low = 0, high = size - 1;

int result = -1;

while (low <= high) {

printActiveArray(arr, low, high);

int mid = low + (high - low) / 2;

if (arr[mid] == target) {

result = mid;

high = mid - 1;

} else if (arr[mid] > target) {

high = mid - 1;

} else {

low = mid + 1;

}

}

return result;

}

int main() {

int arr[] = {1, 3, 3, 3, 7, 9, 9, 15, 17, 19};

int size = sizeof(arr) / sizeof(arr[0]);

int target;

cout << "Enter the target value to search for: ";

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 in the array." << endl;

}

return 0;

}

**Question 3**

#include <iostream>

using namespace std;

void printActiveArray(int arr[], int low, int high) {

cout << "Current active items: [";

for (int i = low; i <= high; ++i) {

cout << arr[i];

if (i != high) cout << ", ";

}

cout << "]" << endl;

}

int findLastOccurrence(int arr[], int size, int target) {

int low = 0, high = size - 1;

int result = -1;

while (low <= high) {

printActiveArray(arr, low, high);

int mid = low + (high - low) / 2;

if (arr[mid] == target) {

result = mid;

low = mid + 1;

} else if (arr[mid] > target) {

high = mid - 1;

} else {

low = mid + 1;

}

}

return result;

}

int main() {

int arr[] = {1, 3, 3, 3, 7, 9, 9, 15, 17, 19};

int size = sizeof(arr) / sizeof(arr[0]);

int target;

cout << "Enter the target value to search for: ";

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 in the array." << endl;

}

return 0;

}

**Question 4**

#include <iostream>

using namespace std;

void printActiveArray(int arr[], int low, int high) {

cout << "Current active items: [";

for (int i = low; i <= high; ++i) {

cout << arr[i];

if (i != high) cout << ", ";

}

cout << "]" << endl;

}

int findFirstOccurrence(int arr[], int size, int target) {

int low = 0, high = size - 1;

int result = -1;

while (low <= high) {

printActiveArray(arr, low, high);

int mid = low + (high - low) / 2;

if (arr[mid] == target) {

result = mid;

high = mid - 1;

} else if (arr[mid] > target) {

high = mid - 1;

} else {

low = mid + 1;

}

}

return result;

}

int findLastOccurrence(int arr[], int size, int target) {

int low = 0, high = size - 1;

int result = -1;

while (low <= high) {

printActiveArray(arr, low, high);

int mid = low + (high - low) / 2;

if (arr[mid] == target) {

result = mid;

low = mid + 1;

} else if (arr[mid] > target) {

high = mid - 1;

} else {

low = mid + 1;

}

}

return result;

}

int countOccurrences(int arr[], int size, int target) {

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

if (firstIndex == -1) {

return 0;

}

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

return lastIndex - firstIndex + 1;

}

int main() {

int arr[] = {1, 3, 3, 3, 7, 9, 9, 15, 17, 19};

int size = sizeof(arr) / sizeof(arr[0]);

int target;

cout << "Enter the target value to search for: ";

cin >> target;

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

if (occurrences > 0) {

cout << "The target value occurs " << occurrences << " time(s) in the array." << endl;

} else {

cout << "The target value is not found in the array." << endl;

}

return 0;

}