

Santa Monica College
CS 20A: C++ Data Structures
Assignment 12: Searching Lists: Sequential, Binary, and Fibonacci Searches

In this assignment, you will implement three search algorithms: sequential, binary, and Fibonacci.

Please download the starter files for this assignment from the Files tab. For each algorithm, provide an explanation of its time complexity. **Do not alter the function definition or driver code in any way.** Programs that crash are subject to a 50% penalty. PLEASE NOTE: You may not use any Standard Template Library (STL) classes for this assignment.

1. Implement the Sequential Search Algorithm

```
template<class T>
int sequential_search (T array[], int size, T value);
```

The sequential search algorithm finds the location of an item in an array, sorted or unsorted.

2. Implement the Recursive and Non-Recursive Binary Search Algorithms

```
template<class T>
int binary_searchR(T array[], int first, int last, T value);

template<class T>
int binary_searchNR(T array[], int first, int last, T value);
```

The binary search algorithm finds the location of an item in a sorted array. The binary search algorithm compares a search key to the middle element in a sorted array.

1. If the search key equals the middle element, return the index of the element.
2. If the search key is greater than the middle element, continue searching in the right half of the array.
3. If the search key is less than the middle element, continue searching in the left half of the array.
4. If the search key is not found (first > last), return -1.

3. Implement the Fibonacci Search Algorithm

```
template<class T>
int fibonacci_search (T array[], int size, T value);
```

Let k be defined as an element in F , the array of Fibonacci numbers. $n = F_m$ is the array size. If the array size is not a Fibonacci number, let F_m be the smallest number in F that is greater than n .

The array of Fibonacci numbers is defined where $F_{k+2} = F_{k+1} + F_k$, when $k \geq 0$, $F_1 = 1$, and $F_0 = 0$.

To test whether an item is in the list of ordered numbers, follow these steps:

1. Set $k = m$.
2. If $k = 0$, stop. There is no match; the item is not in the array.
3. Compare the item against element in F_{k-1} .
4. If the item matches, stop.
5. If the item is less than entry F_{k-1} , discard the elements from positions $F_{k-1} + 1$ to n . Set $k = k - 1$ and return to step 2.
6. If the item is greater than entry F_{k-1} , discard the elements from positions 1 to F_{k-1} . Renumber the remaining elements from 1 to F_{k-2} , set $k = k - 2$, and return to step 2.

(Excerpt from Wikipedia, "Fibonacci search technique," https://en.wikipedia.org/wiki/Fibonacci_search_technique, 7/27/2015)