Fibonacci Search

Fibonacci search is derived from Golden section search, an algorithm by Jack Kiefer (1953) to search for the maximum or minimum of a uni-modal function in an interval.

Fibonacci Numbers are recursively defined as F(n) = F(n-1) + F(n-2), F(0) = 0, F(1) = 1.
First few Fibonacci Numbers are 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ... n

Differences with Binary Search:

- Fibonacci Search divides given array in unequal parts.
- Binary Search uses division operator to divide range. Fibonacci Search doesn't use /, but uses + and -. The division operator may be costly on some CPUs.

  > Fibonacci search divides the array into two parts that have sizes that are consecutive Fibonacci numbers. On average, this leads to about 4% more comparisons to be executed, but it has the advantage that one only needs addition and subtraction to calculate the indices of the accessed array elements, while classical binary search needs bit-shift, division or multiplication operations that were less common at the time Fibonacci search was first published.

- Fibonacci Search examines relatively closer elements in subsequent steps. So when input array is big that cannot fit in CPU cache or even in RAM, Fibonacci Search can be useful.

Fibonacci search has an average- and worst-case complexity of O(log n)

Let arr[0..n-1] be the input array and element to be searched be x

- Find the smallest Fibonacci Number greater than or equal to n. Let this number be fibM [m^{th} Fibonacci Number]. Let the two Fibonacci numbers preceding it be fibMm1 [$(m-1)^{th}$ Fibonacci Number] and fibMm2 [$(m-2)^{th}$ Fibonacci Number].
- While the array has elements to be inspected:

```
Compare x with the last element of the range covered by fibMm2.
If x matches,
  return index.
Else If x is less than the element,
  move the three Fibonacci variables two Fibonacci down, indicating elimination of approximately rear two-th
Else x is greater than the element,
  move the three Fibonacci variables one Fibonacci down. Reset offset to index. Together these indicate elim
```

- Since there might be a single element remaining for comparison, check if fibMm1 is 1. If Yes, compare x with that remaining element. If match, return index.