Practical 4

Title: Searching algorithm.

Problem statement
a) Write a python program to store roll number of student in array who atten ded training program in random order Write a program whether particular student attend the program or not.
b) Write a program to store roll numbers of students array who attended training program in sorted order. Write function for searching whether particular student attended program or not using binary & fibonacci search.

Learning Objectives.
a) To study what is searching.
b) To understand implementation of various searching algorithms.

Theory.
1) Linear Search
Linear search is very easy to implement, in linear search we can traverse on each element, so it is takes more time than the other searching algorithms. It has high time complexity than any other searching algorithm.

b) Fibonacci search.

Fibonacci search is comparison based algorithm that uses fibona number to search an element in a sorted in a binary search.

c) sentinal search:

To reduce over head of checking the list length the value to be search can be appended to the list attended as a sentinal value is one where present the termination of a loop that process the after data.

d) Binary search.

- Binary search is used with sorted array in a binary search we follow the following step.

*) first we have sorted array then we divide into two equal parts.

* Algorithm:

a) linear search:

step 1) Ask roll no.

2) check roll no in array.

3) If it is present (linear) then print.

4) EndIf.

b) Sentinal Search.

step 1) take sentinel & add it at last position of array.

step 2) Sentinel = key.

3) If position == key print.

4) position = at end of list

5) if index x <= n present

6) Else. not present.

7) EndIf.

c) Binary search

step 1) Accept key (element to search)

step 2) Find mid = (low + high) / 2

step 3) Compare A[mid] with key

a) If both are matching stop & print position of key.

b) If key is greater than A(mid) low = mid + 1 & if low <= high go to step 2. else go to step 4

Step 4) print that element is not found.

d) Fibonacci search.

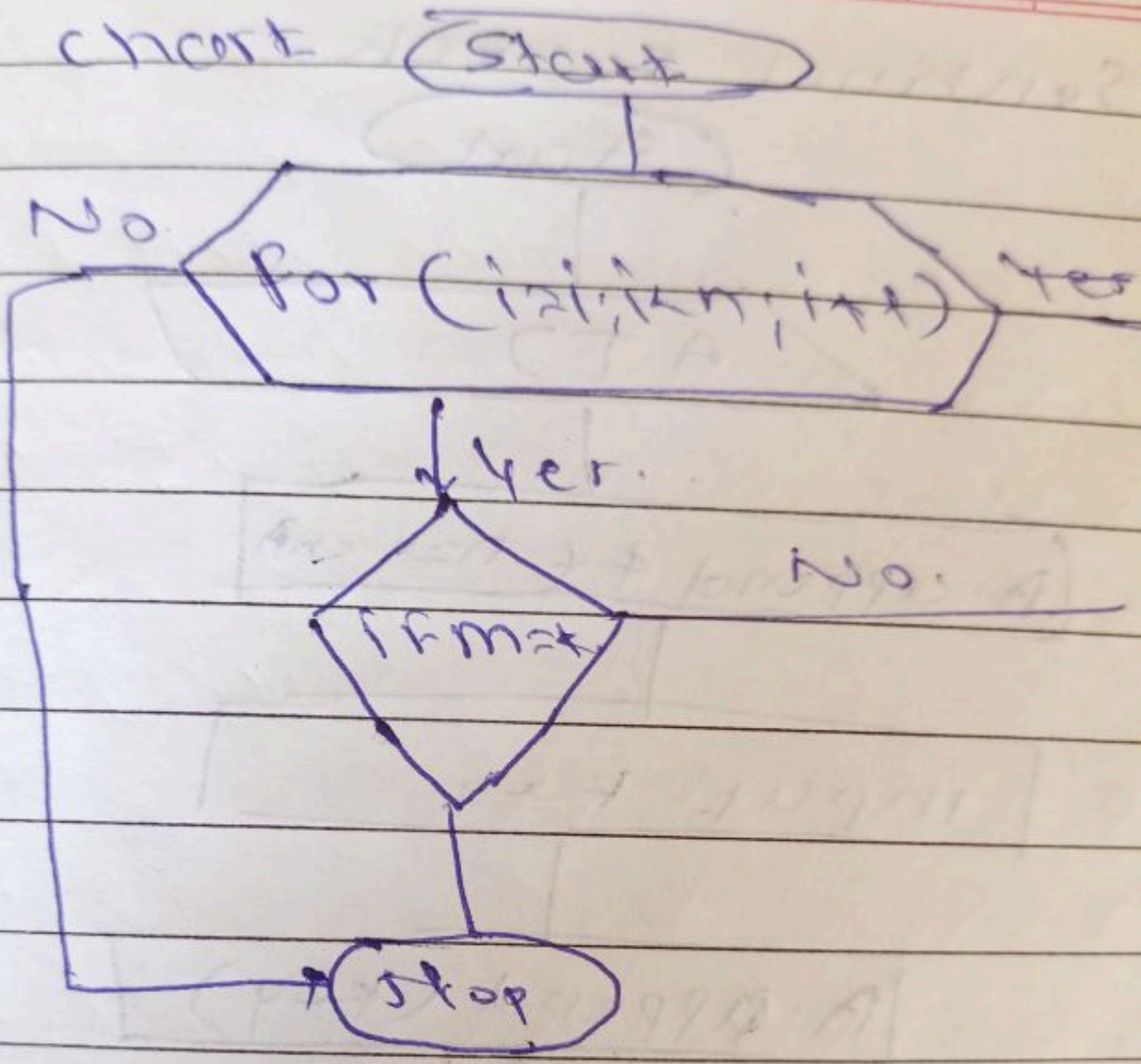
step 1) A: sorted array.
c: key integer.
fib1 = 1 , fib2 = 1 fib = 2.

step 2) while = fib < n
fibd = fib1
fib1 = fib
fib = fib1 + fib2.
end while.

step 3) i = 0 offset = 0
while (fib > 1) do.
i = min(offset + fib2, n)
fib = fib1
fib1 = fib2
fib2 = fib - fib1, offset = i
end if A(i) > c then.
fib = fib2.
fib1 = fib1 - fib2.

* Software requirement.
- Python 3, IDE like VS code

Flow chart



2) Sentinel Search