Practice 05:

Implementation of Fibonacci Search

Code:

def fibMonaccianSearch(arr, x, n):

fibMMm2 = 0 # (m-2)'th Fibonacci No.

fibMMm1 = 1 # (m-1)'th Fibonacci No.

fibM = fibMMm2 + fibMMm1 # m'th Fibonacci

while (fibM < n):

fibMMm2 = fibMMm1

fibMMm1 = fibM

fibM = fibMMm2 + fibMMm1

offset = -1

while (fibM > 1):

# Check if fibMm2 is a valid location

i = min(offset+fibMMm2, n-1)

# If x is greater than the value at index fibMm2, cut the subarray array from offset to i

if (arr[i] < x):

fibM = fibMMm1

fibMMm1 = fibMMm2

fibMMm2 = fibM - fibMMm1

offset = i

# If x is less than the value at index fibMm2, cut the subarray after i+1

elif (arr[i] > x):

fibM = fibMMm2

fibMMm1 = fibMMm1 - fibMMm2

fibMMm2 = fibM - fibMMm1

# if element found then return index

else:

return i

# comparing the last element with x

if(fibMMm1 and arr[n-1] == x):

return n-1

# if element not found then return -1

return -1

# Main Function

arr = [10, 22, 35, 40, 45, 50,

80, 82, 85, 90, 100,235]

n = len(arr)

x = 235

ind = fibMonaccianSearch(arr, x, n)

if ind>=0:

print("Found at index:",ind)

else:

print(x,"isn't present in the array");