//DSL Lab 04 Searching

```
#input function
def in_student():
    n =int(input("How many students attended training program: "))
    student_lis =[]
    for i in range(n):
        roll = int(input("Enter roll number of "+str(i+1)+"/"+str(n)+": "))
        student_lis.append(roll)
    #print(student_lis)
    return student_lis

#linear search
def linear_search(student_lis, n):
    status = []
    for i in student_lis:
        if(i==n):
            status.append('p')

            break
    status = set(status)
    print(student_lis)
    if(status=={'p'}):
        print("Student of roll number "+str(n)+" attended the session! And
index is:"+str(student_lis.index(n)))
    else:
         print("Student of roll number "+str(n)+" was absent in the
session!")

#sentinel search
def sentinel_search(student_lis,n):
    student_lis.append(n)
    N = len(student_lis)
    status = []
    for i in range(N-1):
        if(student_lis[N-1] == student_lis[i]):
            status.append('p')

            break
    status = set(status)
    student_lis.remove(n)
    print(student_lis)
    if(status=={'p'}):
        print("Student of roll number "+str(n)+" attended the session! And
index is:"+str(student_lis.index(n)))
    else:
         print("Student of roll number "+str(n)+" was absent in the
session!")

#binary search
def binary_search(student_lis, n):
    student_lis = sorted(student_lis)
```

```
    low = 0
    high = len(student_lis) - 1
    mid = 0
    status = []
    while low <= high:
        mid = (high + low) // 2
        if student_lis[mid] < n:
            low = mid + 1
        elif student_lis[mid] > n:
            high = mid - 1
        else:
            status.append('p')
            break

    status = set(status)
    print(student_lis)
    if(status == {'p'}):
          print("Student of roll number "+str(n)+" attended the session! And
index is:"+str(student_lis.index(n)))
    else:
          print("Student of roll number "+str(n)+" was absent in the
session!")


#other method
# 24 in [24,65,12,20,52,35,3,58]

# [24,65,12,20,52,35,3,58].index(24)


# def fibo_series(n):

#     if n<=1:
#         return n
#     else:
#         return fibo_series(n-1)+fibo_series(n-2)
# fibo_list = []
# nterm = 20
# if nterm<=0:
#     print("Enter positive number!")
# else:
#     for i in range(nterm):
#        fibo_list.append(fibo_series(i))
#     print(fibo_list)


#fibonacci search
def fibonacci_search(student_lis, n):
    student_lis = sorted(student_lis)
    N = len(student_lis)
    offset = -1
    status = []
    f0 = 0
    f1 = 1
    f2 = 1
    while(f2 < N):
        f0 = f1
```

```
        f1 = f2
        f2 = f1 + f0
    while(f2 > 1):
        i = min(offset + f0, N - 1)
        if student_lis[i] < n:
            f2 = f1
            f1 = f0
            f0 = f2 - f1
            offset = i
        elif student_lis[i] > n:
            f2 = f0
            f1 = f1 - f0
            f0 = f2 - f1
        else:
            status.append('p')
            break
    if (f1) and (student_lis[N - 1] == n):
        status.append('p')

    status = set(status)
    print(student_lis)
    if(status == {'p'}):
          print("Student of roll number "+str(n)+" attended the session! And
index is:"+str(student_lis.index(n)))
    else:
          print("Student of roll number "+str(n)+" was absent in the
session!")


student_lis = in_student()
print()

otp = 24

while(otp !=0):
    n = int(input("Enter roll to search: "))
    print()
    print(" MENU ".center(30,"~"))
    print("1. Linear Search\n2. Sentinel Search\n3. Binary Search\n4.
Fibonacci Search")
    otp = int(input("Enter an option to select method (0 to exit): "))
    if(otp==1):
        linear_search(student_lis, n)
    elif(otp==2):
        sentinel_search(student_lis, n)
    elif(otp==3):
        binary_search(student_lis, n)
    elif(otp==4):
        fibonacci_search(student_lis, n)
    else:
        print("Program ended successfully!")
```