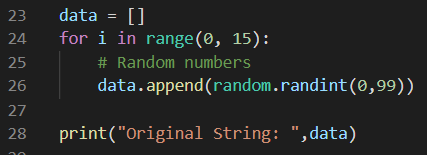
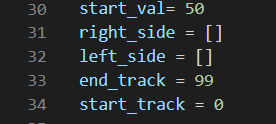
Shivraj Naorem

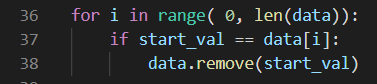
212010054

Look Disk Scheduling Algorithm

Code:

For loop to generate and store the random numbers for our main argument string.

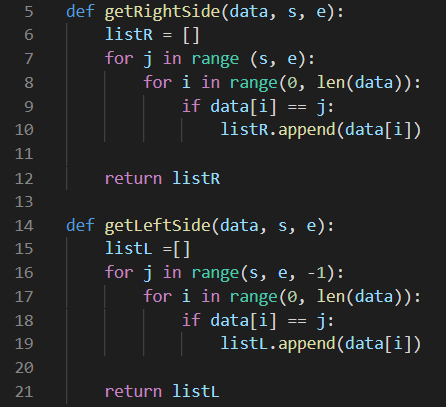
****Starting value to start the algorithm and to initialize the start and end of the track. Variables at 31 and 32 to store the values that are above and below the start value in the main string respectively.

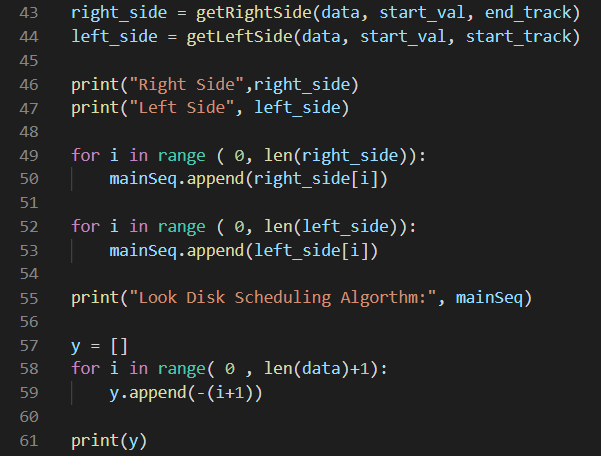


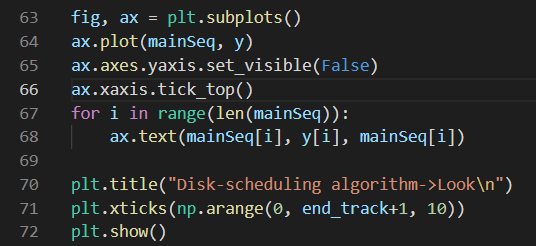
To deal with condition where the start value is already in the main string.



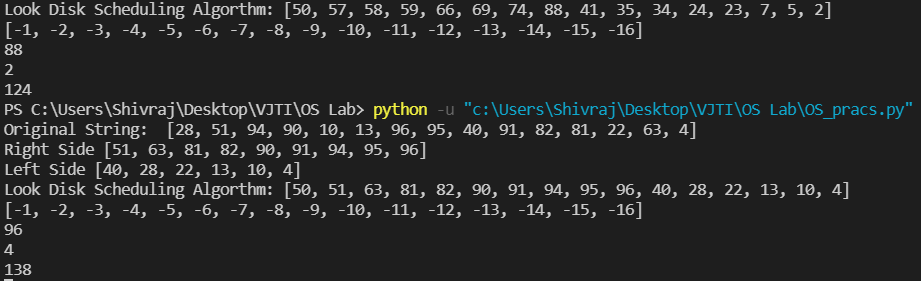
To initialize the variable that will store the sequence of the algorithm.

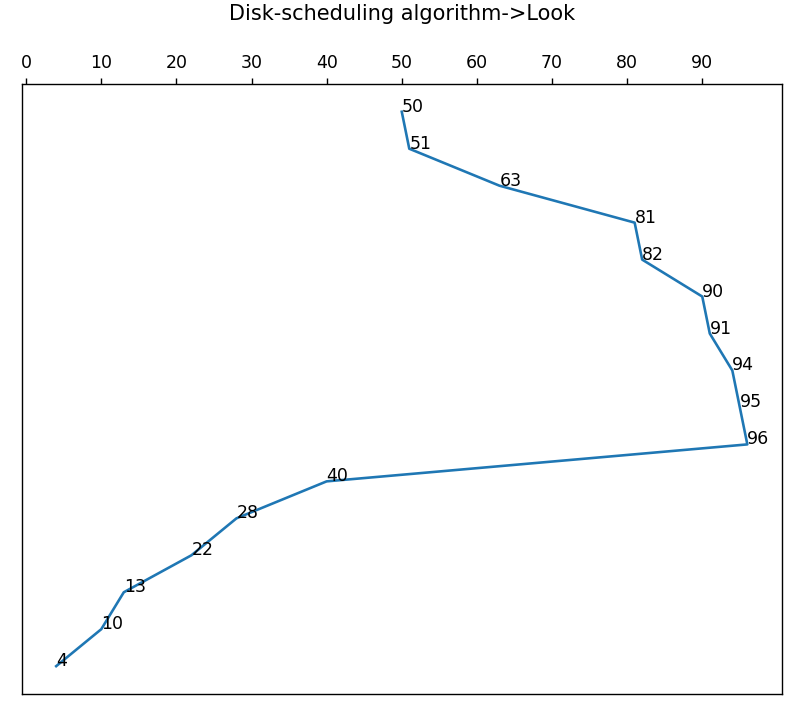
Functions to store the values above and below the starting value respectively and returning it.

We will be moving to the right first so the right sequence will be appended first then the left to the algorithm sequence. And now initializing the Y-axis values for plotting the graph.

For Plotting the graph.

Output:



Graph:

Full Code:

import matplotlib.pyplot as plt

import numpy as np

import random

def getRightSide(data, s, e):

    listR = []

    for j in range (s, e):

        for i in range(0, len(data)):

            if data[i] == j:

                listR.append(data[i])

    return listR

def getLeftSide(data, s, e):

    listL =[]

    for j in range(s, e, -1):

        for i in range(0, len(data)):

            if data[i] == j:

                listL.append(data[i])

    return listL

data = []

for i in range(0, 15):

    # Random numbers

    data.append(random.randint(0,99))

print("Original String: ",data)

start\_val= 50

right\_side = []

left\_side = []

end\_track = 99

start\_track = 0

for i in range( 0, len(data)):

    if start\_val == data[i]:

        data.remove(start\_val)

mainSeq = []

mainSeq.append(start\_val)

right\_side = getRightSide(data, start\_val, end\_track)

left\_side = getLeftSide(data, start\_val, start\_track)

print("Right Side",right\_side)

print("Left Side", left\_side)

for i in range ( 0, len(right\_side)):

    mainSeq.append(right\_side[i])

for i in range ( 0, len(left\_side)):

    mainSeq.append(left\_side[i])

print("Look Disk Scheduling Algorthm:", mainSeq)

y = []

for i in range( 0 , len(data)+1):

    y.append(-(i+1))

print(y)

print( max(data))

print(min(data))

totalStepsMoved = (max(data)-start\_val) + (max(data)-min(data))

print(totalStepsMoved)

fig, ax = plt.subplots()

ax.plot(mainSeq, y)

ax.axes.yaxis.set\_visible(False)

ax.xaxis.tick\_top()

for i in range(len(mainSeq)):

    ax.text(mainSeq[i], y[i], mainSeq[i])

plt.title("Disk-scheduling algorithm->Look\n")

plt.xticks(np.arange(0, end\_track+1, 10))

plt.show()