num1=100

num2=200

num3=6

if(5>=num3):

if(num1>100 or num2>150):

print("1")

elif(num1>=100 and num2>150):

print("2")

else:

print("3")

Answer - 2

//======================================================

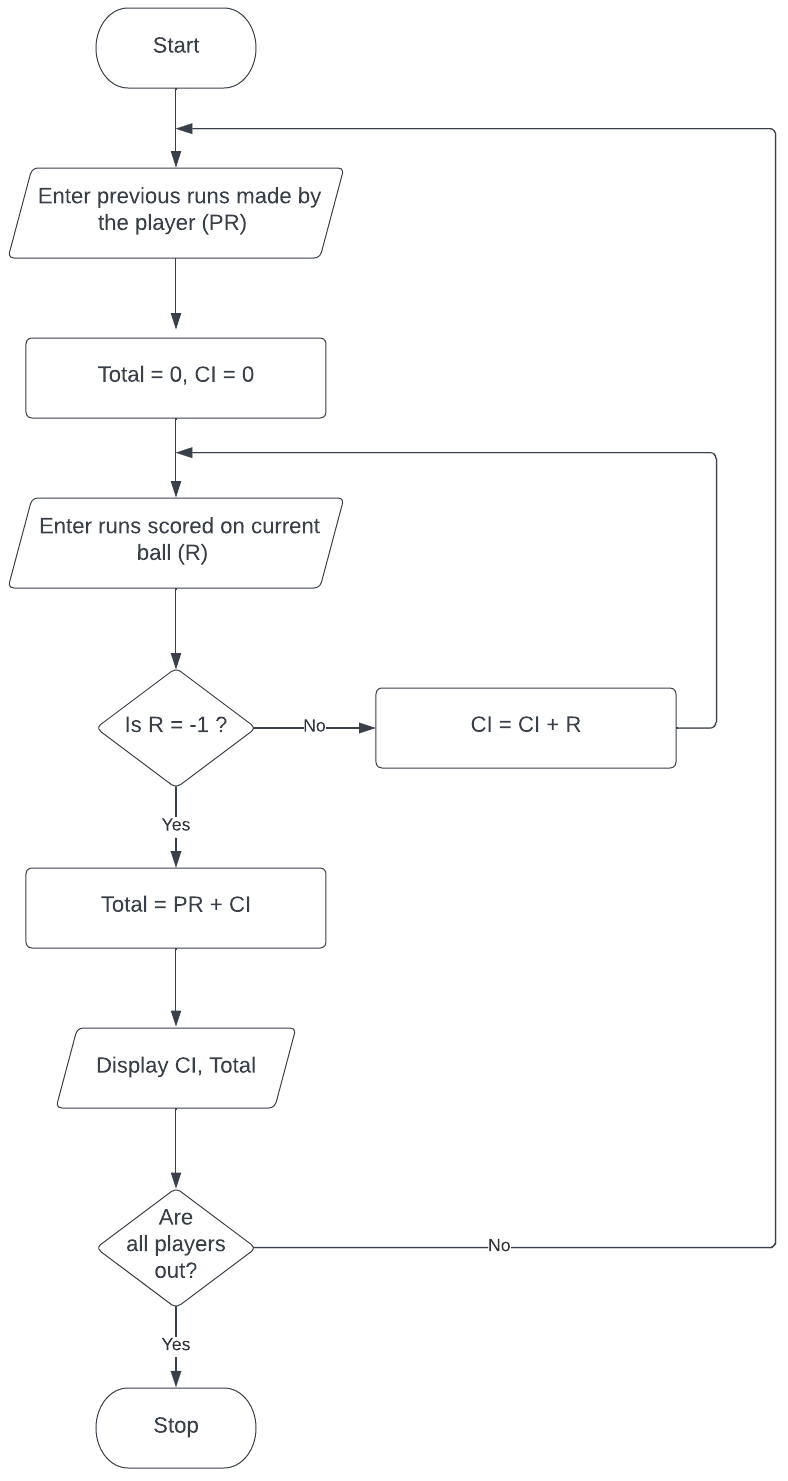
Question 6

In a cricket match, a viewer carries a laptop in the stadium every day to keep record of the runs made by each batsman.

When a player comes to bat, the viewer enters the previous runs made by the player. Further, he enters the runs scored for each ball.

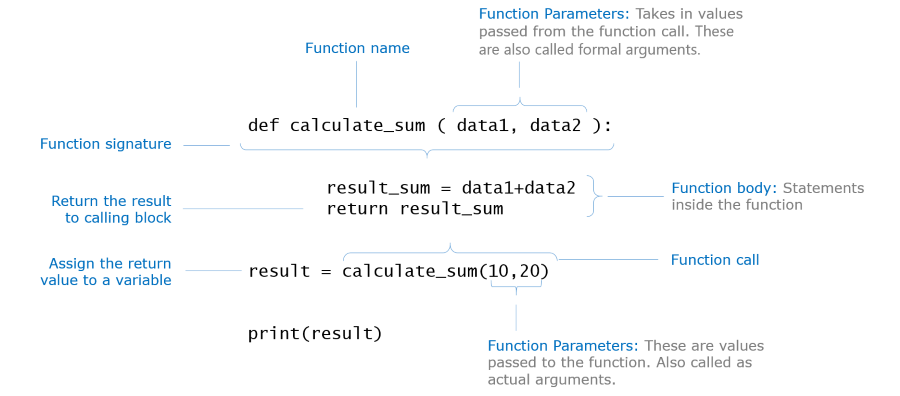
When the runs entered on a ball is -1, it means the player is out. The laptop then displays the runs made by the player in the current innings as well as the total.

The process continues until all the players are out. Draw a flowchart for the task stated above.



//================================

function in program -

A function is a block of code that performs a particular task. In python, functions are declared using the keyword def.  
  
  


def add(a,b):

    c=a+b

    return c

res=add(10,20)

print(res)

//------------------------------------------------------------

#Define a function which check given number is even or od

def isEven(num):

    if num%2==0:

        return True

    else:

        return False

#Calling function

res=isEven(4)

print(res)

//=====================================================  
  
Display list -

numbers=[10,20,30,40,50]

print(numbers[2])

#size of list

print(len(numbers))

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

    print(numbers[i])

//-------------------------------------------  
Q. create a list and find the square of each element if list  
 st1=[1,2,3,4,5,6]  
for i in range(0,len(list1)):  
    print(list1[i]\*list1[i])

//------------------------------------------------

Linear Search -

def linear\_search(list,num):

    flag=False

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

if list[i]==num:

            flag=True

            break

    return flag

list=[10,20,40,300,56]

n=56

res=linear\_search(list,n)

if res==True:

    print("Element Found")

else:

print("Element Not Found")  
  
//--------------------------------------------------------------------------------------  
Binary Search -

def binary\_search(list,num):

flag=False

#index pos of starting element in list

low=0

#index pos of last element in list

high=len(list)-1

mid=0

while(low<=high):

#middle element pos

mid=(low+high)//2

#if search element is greater than middle

if num>list[mid]:

low=mid+1

elif num<list[mid]:

high=mid-1

else:

flag=True

break

return flag  
//================================================  
  
Q1 of 3outlined\_flag

Using the binary search strategy of having numbers in sorted order, if you have to find 25 from a list containing numbers from 1 to 50 arranged in ascending order,how many guesses do you have to make?

0

1

25

50

Answer - 1  
-------------------------------------  
Q2 of 3outlined\_flag

Using the binary search strategy of having numbers in sorted order, if you have to find 50 from a list containing numbers from 1 to 50 arranged in ascending order, how many guesses do you have to make?

6

8

4

2  
Answer - 6

--------------------------------------  
Q3 of 3outlined\_flag

Do you think the number of guesses to be made is equal to the position of the number to be guessed?

Yes

No

Answer - No  
  
//---------------------------------------------------------------------------------  
Problem Statement

The flight ticket rates for a round-trip (Mumbai->Dubai) were as follows:   
Rate per Adult: Rs. 37550.0   
Rate per Child: 1/3rd of the rate per adult   
Service Tax: 7% of the ticket amount (including all passengers)   
As it was a holiday season, the airline also offered 10% discount on the final ticket cost (after inclusion of the service tax).  
Find and display the total ticket cost for a group which had adults and children.  
  
Test the program with different input values for number of adults and children.  
Test the program with different input values for number of adults and children.

|  |  |  |
| --- | --- | --- |
| **Sample Input** | | **Expected Output** |
| Number of adults | Number of children |  |
| 5 | 2 | Total Ticket Cost: 204910.35 |
| 3 | 1 | Total Ticket Cost: 120535.5 |

Code -   
def calculate\_total\_ticket\_cost(no\_of\_adults, no\_of\_children):  
    total\_ticket\_cost=0  
    #Write your logic here  
    intermediate\_result=(no\_of\_adults\*37550.0)+(no\_of\_children\*(37550/3))  
    service\_tax=(7/100)\*intermediate\_result  
    intermediate\_result=intermediate\_result+service\_tax  
    discount=(10/100)\*intermediate\_result  
    total\_ticket\_cost=intermediate\_result-discount  
    return total\_ticket\_cost

//============================================================

 Sorting Algorithms -

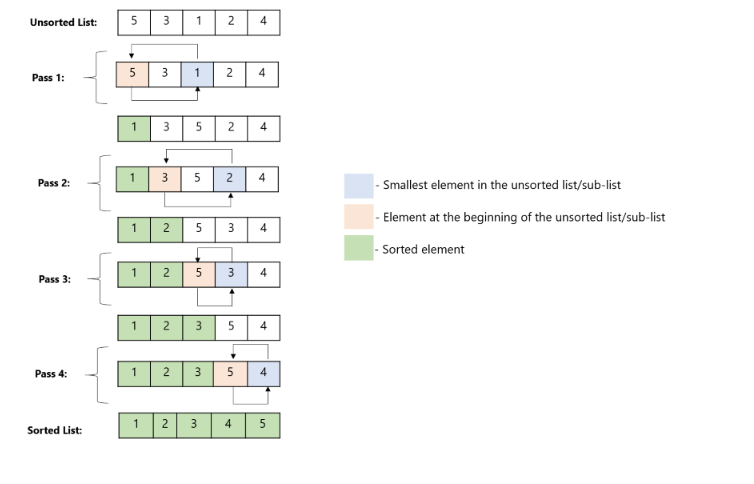
1. Selection sort -

The algorithm you have seen in the animation is as follows:

Assume that we have an unsorted list.

1. Find the smallest number and swap it with the number at the first position in the list.
2. Find the next smallest number and swap it with the number at the second position in the list.
3. Find the next smallest number and swap it with the number at the third position in the list.
4. Continue doing it until you are done with the number at the last position in the list.

This strategy of sorting is known as **Selection Sort**



//==============================

Q1 of 1outlined\_flag

You have just learnt the selection sort algorithm to sort a list of elements in ascending order.What changes should be made in the algorithm to sort the elements in descending order?

Find the smallest element and swap it with the element at the first position in the listFind the largest element and swap it with the element at the first position in the list checkUse the same algorithm, sort the elements in ascending order, finally reverse the elements in the list Find the largest element and swap it with the element at the last position in the list

//===========================================

Q1 of 1

You have just learnt the selection sort algorithm to sort a list of elements in ascending order.  
What changes should be made in the algorithm to sort the elements in descending order?

1. Find the smallest element and swap it with the element at the first position in the list  
   B.Find the largest element and swap it with the element at the first position in the list   
   C.Use the same algorithm, sort the elements in ascending order, finally reverse the elements in the list  
   D.Find the largest element and swap it with the element at the last position in the list

Answer - B

//======================================  
Q.) Write a python program that displays a message as follows for a given number:

1. If it is a multiple of three, display "Zip"
2. If it is a multiple of five, display "Zap".
3. If it is a multiple of both three and five, display "Zoom".
4. If it does not satisfy any of the above given conditions, display "Invalid".

Code -

def display\_message(number):

if number % 3 == 0 and number % 5 == 0:

print("Zoom")

elif number % 3 == 0:

print("Zip")

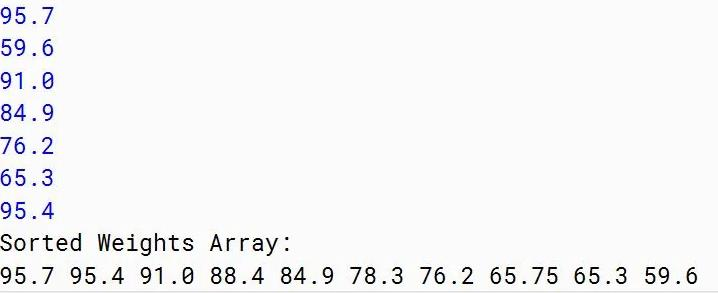
elif number % 5 == 0:

print("Zap")

else:

print("Invalid")  
//===================================

**Q.) Write a program to input and sort the weight of ten people. Sort and display them in descending order using the selection sort technique.**



Code -   
def weights(arr):

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

min=arr[i]

pos=i

for j in range(i+1,len(arr)):

if arr[j]> min:

min=arr[j]

pos=j

arr[pos]=arr[i]

arr[i]=min

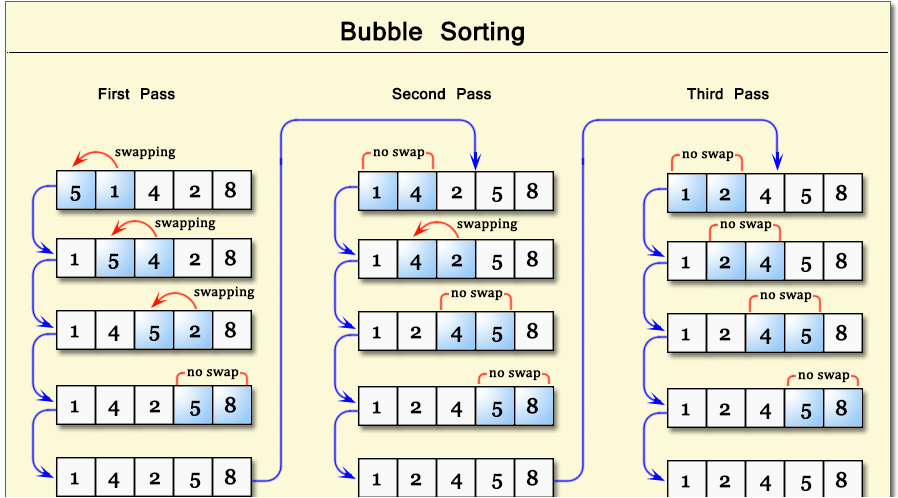
return arr

sort\_arr= weights([95.7,59.6,91.0,84.9,76.2,65.3,95.4])

print(sort\_arr)

//=============================

2 ) Bubble sort -



Q.)Consider the list of numbers given below:  
num\_list = [ 67,34,8,22,23]  
How many passes bubble sort algorithm will go through to sort the numbers in the above list in ascending order?

1

3

4

7

Answer - 4  
//----------------------------------------------------------------

1. Consider the list of numbers given below:  
   num\_list = [89,43,99,55,87,67] ,How many passes bubble sort algorithm   
   Answer - 4  
   code -  
   def bubble\_sort(num\_list):

n = len(num\_list)

for i in range(n):

for j in range(0, n-i-1):

if num\_list[j] > num\_list[j+1]:

num\_list[j], num\_list[j+1] = num\_list[j+1], num\_list[j]

num\_list = [89, 43, 99, 55, 87, 67]

bubble\_sort(num\_list)

print("Sorted list:", num\_list)

//---------------------------------------------------------------------------

Problem Statement

Mary is a kindergarten teacher. She has given a task to the children after teaching them a list of words. The task is to find the unknown words (other than the words they already know) from the given text. Write a python function which accepts the text and the known list of words and returns the set of unknown words found.

Return -1 if there are no unknown words.

|  |  |
| --- | --- |
| **Sample Input** | **Expected Output** |
| Text: "the sun rises in the east" Vocabulary: ["sun","in","east","doctor","day"] | {'rises', 'the'} |

//==============================  
Write a Python program to find the sum of digits of a given number. E.g. Sum of number 123 will be 6  
Note: Initialize the number with various values and test your program.  
//==================================================

Let’s compare selection sort and bubble sort algorithms in this exercise.

Combine the selection sort and bubble sort programs as per the template code provided below and display the number of passes for each of them.

Invoke both the functions (selection\_sort() and bubble\_sort()) using the following two lists and observe the results.

Case 1: [8,2,19,34,23, 67, 91]

Case 2: [91,8,19,23,34,67,2]

Code in Python 3

#lex\_auth\_0127667385791856643328

def swap(num\_list, first\_index, second\_index):

    #Remove pass and copy the code earlier written for this function

    pass

def find\_next\_min(num\_list,start\_index):

    #Remove pass and copy the code earlier written for this function

    pass

def selection\_sort(num\_list):

    #Remove pass and copy the code earlier written for this function

    #Modify it to return the total number of passes the algorithm has gone through to sort the list

    pass

def bubble\_sort(num\_list):

    total\_no\_of\_passes=0

    end\_index=len(num\_list)

    for index1 in range(0, end\_index-1):

//=================