**Question 1:** Read the following python code carefully:

item\_number = []

item\_name = []

item\_description = []

item\_reservedPrice = []

item\_bids = []

item\_numberOfBidsCount = []

item = int(input("How many items to be sold: "))

while item <10:

item = int(input("Minimum 10 items. How many items to be sold: "))

item\_bids = [0 for i in range(0,item)]

item\_numberOfBidsCount = [0 for i in range(0,item)]

for i in range(0,item):

number = int(input("Enter unique item number: "))

/\*\*\*\*

Missing Line of Codes

\*\*\*\*/

item\_number.append(number)

name = input("Enter item name: ")

item\_name.append(name)

description = input("Enter item description: ")

item\_description.append(description)

reservedPrice = float(input("Enter item reserved price: "))

item\_reservedPrice.append(reservedPrice)

1. Complete the missing portion of the above code. [1]

Answer:…………………………………………………………………......………………………

……………………………………………………………………………...…………….…………

………………………………………………………………………………………………………

………………………………………………………………………………………………………………………..……………….……….……………………………………………………………………………..………………………………………………………………………….………….

1. Explain how will your program work? [2]

Answer: ..……………………………………………………………………………….……..……

……..………………………………………………………………………………………………..…………………………….……………..……….…………………………………………………………………………………………………………………………………………………………

……………………………………….……….………..……………………………………………………………………………………………………..…………………………………………….

………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

1. Draw flowchart for above program. [4]

Answer:

**Question 2:** In Pre-Release Task 2 you have to check if the item number say ***sampleNumber*** is exist or not which is stored in **item\_number[i]**. If item number is existing you need to take input a price say ***buyerPrice*** with a buyer id say ***buyerID*** and check it with the item reserved price stored at **item\_reservedPrice[i]**. If the stored reserved price is less than buyer price then increases the bid count at **itemBidCount[i]** by 1. If item number is absent then give an output with the text “***I am extremely sorry! Nothing found in the container.***”

Solve this problem with pseudocode or python or flow chart and don’t forget to initialize the value where it is needed. [6]

Answer: …………………………………………………………………………………………….

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

**Question 3:** Python codes written below is the solve of Task 3.

total\_amount = 0

total\_sold = 0

total\_unsold = 0

total\_noBid = 0

sold = []

unsold = []

noBid = []

for i in range (0,item):

if (item\_numberOfBidsCount[i]>0): # if condition 1

if(item\_bids[i]>item\_reservedPrice[i]): # if condition 2

sold.append(item\_number[i])

total\_amount = total\_amount + item\_bids[i]

total\_sold = total\_sold + 1

else: # else condition 1

unsold.append(item\_number[i])

total\_unsold = total\_unsold + 1

else: # else condition 2

noBid.append(item\_number[i])

total\_noBid = total\_noBid + 1

print("Total Cost: ",(total\_amount+total\_amount\*0.1))

print("Number of Sold Item: ",(total\_sold))

print("Number of Unsold Item: ",(total\_unsold))

print("Number of Nobid Item: ",(total\_noBid))

print("Sold Items: ",(sold))

print("Unsold Items: ",(unsold))

print("Items with no bid: ",(noBid))

1. Explain both if conditions 1 and 2 and say if there any change is needed? [2]+[2]

Answer: …………………………………………………………………………………………….

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

1. Explain both else conditions 1 and 2 and say if we do not put any else condition [2]+[2]

what will happen?

Answer: …………………………………………………………………………………………….

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

**Question 4:** In Question no 1 in the given python code we are using several array/list.

1. What is purpose of **item\_bids[]** and **item\_numberOfBidsCount[]?** [2]

Answer: ….…………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

1. Explain the line of codes regarding these two arrays/lists. [2]

Answer: ……………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………