CO1107 Test 1 Answer

Question 1

2 points

During binary search, we maintain a "low" index and a "high" index. There is a "target" value that we are trying to find. There is also a "midpoint" which is simply (low+high)//2.

Suppose we find that the value in the array at the midpoint is greater than the target value. We need to adjust either low or high. Which one of the following options is the adjustment we need to make?

- (A) low=mid
- (B) high=mid
- (C) low=mid+1
- (D) high=mid+1
- (E) high=mid-1 — Correct answer

Question 2

4 points

What will be the output of the following code?
Consider the following:
append (aQueue , A): will append A into the queue aQueue
Serve(aQueue): will serve from the queue aQueue
Size (aQueue): will return the number of items in the queue aQueue

```
def doSomething(aQueue):
    n=Size(aQueue)
    theMax=aQueue[2]
    for counter in range(n, 0,-1):
        item=Serve(theMax)
        if item>theMax:
            theMax=2.5 * item
            append (aQueue,theMax)

    return theMax

q=[15,2,-7,16,8,22]
print(doSomething(q))
```

- (A) 37.5 — Correct answer
- (B) 6
- (C) -10.5
- (D) -7.5

Question 4

1.5 points

Consider a situation where swap operation is very costly. Which of the following sorting algorithms should be preferred so that the number of swap operations are minimized in general?

- (A) Selection sort
- (B) Insertion sort — Correct answer
- (C) Bubble sort

Question 5

2 points

What will be the output of the following code?

```
def doSomething (numList ):
    test = [0] * 5
    for aNum in numList :
        test [aNum] = test[aNum] + 1
    return test

aList=[4,1,2,1]
print(doSomething(aList))
```

- (A) [0, 2, 1, 0, 1] — Correct answer
- (B) [4, 1, 2, 1, 0]
- (C) [0, 0, 0, 0, 0]
- (D) [1, 1, 1, 1, 1]

Question 6

2 points

What will be the output of the following code?

```
aList=["4","3","2"]
bList=["4","B","6"]
cList=[]

for i in range(len(aList)):
    if i%2==0:
        item=int(aList[i])+int(bList[i])
    else:
        item=int(aList[i])*bList[i]
    cList.append(item)
print(cList)
```

A [8, 'BBB', 8] Correct answer

B [4, 'BBB', 8]

C [8, 'B', 8]

D [8, 'BBB', 12]

Question 7

2 points

what will be the output of the following code ?

```
def doSomething(bList,aFloat):
    bList.append(10)
    aFloat = aFloat*4
aList = [1,5]
aFloat = 5

doSomething(aList,aFloat)
aList.append(aFloat)
print(aList)
```

A [1,5,10,5] Correct answer

B [1,5,5,5]

C [1,5,20,5]

Question 8

2 points

Examine the numerical list (aList) with the sequence [1, 4, 7, 12, 27, 8, 19, 9], where the **first four numbers** have already been sorted using the algorithm provided below.

What will be the state of the list **aList** after the next iteration of the **for loop**?

```
def doSomething_1 (myList,i,j):
    temp = myList[i]
    myList[i] = myList[j]
    myList[j] = temp


def doSomething (aList):
    n = len(aList)
    for k in range(1,n):
        doSomething_2(aList,k)

def doSomething_2 (aList,j):
    while aList[j-1] > aList[j] and j>0:
            doSomething_1(aList,j-1,j)
            j = j-1
```

(A) 1,4,7,12,27,8,19,9 — Correct answer

(B) 1,4,7,8,12,27,19,9

(C) 1,4,7,8,27,12,19,9

(D) 1,4,7,12,8,27,19,9

Question 9

2 points

What will be printed by the **print(code)** statement in the last line?

```
def goSleep(level):
    print("In dream at level",level)
    if level == 3:
        print("Reached level 3")
        code = 42
        print("Got the code, returning it to level",level-1)
        return code
    else:
        print("I am going deeper to level",level+1)
        code = goSleep(level+1)
        print("I am at level",level)
        print("Got the code, returning it to level",level-1)
        return code

code = goSleep(1)
print(code)
```

(A) 42 — Correct answer

(B) None

(C) Error Message

(D) 43

Question 10

1.5 points

What does the following algorithm describe?

```
1. input the list
2. let k be 1
3. while k is less than the size of the list...
  (a) let j be k
  (b) while the first k elements are not sorted...
    i. shift the jth element left one position
    ii. reduce j by 1
  (c) increase k
4. output the list
```

(A) Nothing, it's not a valid algorithm

(B) insertion sort — Correct answer

(C) selection sort

(D) bubble sort

Question 11

2 points

What will be the output of the following code?

```
def doSomething(aList):
    table = []
    if len(aList)==0:
        return False
    value = aList.pop()
    table.append(value)

    while len(aList)>0:
        item = aList.pop()
        if item > value:
            value = item
        table.append(item)

    while len(table)>0:
        aList.append(table.pop())

    return value

print(doSomething([12, 19, 57, 42]))
```

(A) 4

(B) 12

(C) 57 Correct answer

(D) the sum of all the numbers in a given list which is 130

Question 12

2 points

```
Which of the following statements are true
Given the list: L = [12,6,31,66]
I) L.append(40) gives [12,6,31,66,40]
II) L.append(40) gives [40,12,6,31,66,40]
III) L.pop() leaves L as [6,31,66]
IV) L.pop() leaves L as [12,6,31,66]
```

(A) I & II

(B) I & III

(C) I & IV Correct answer

(D) I & III & IV

Question 13

1.5 points

Binary search works only on a sorted list.

True *Correct answer*

False

Question 14

1.5 points

What happens if the base case isn't defined in recursive programs?

(A) Program gets into an infinite loop Correct answer

(B) An exception is thrown

(C) Program runs once