Survey #3

Name: _____

Consider the following code segment:

```
data = [5,-5,10,-10,0]

t = data[4]
for i in range(0,4):
  data[i] = data[i+1]
data[1] = t

print(data)
```

1: At the end of this code segment, the value of data[1] is: _____

2: At the end of this code segment, the value of data[3] is: _____

3: At the end of this code segment, the value of data[5] is: _____

Consider the following code segment:

```
a = 5
b = a
c = ["A","list","of","words"]
d = c

b = 7
c[0] = "The"
d[3] = "something"

print(a)  # statement 1
print(b)  # statement 2
print(c)  # statement 3
print(d)  # statement 4
```

4: The output generate by statement 1 will be: _____

5: The output generated by statement 2 will be: _____

6: The output generated by statement 3 will be: _____

7: The output generated by statement 4 will be: _____

8: [Reasonably Easy] Write a Python function, find_smallest, that takes a list of integers as its parameter. The function should identify the smallest element in the list and return it as the result of the function. Your program should exit with an appropriate error message if the list doesn't contain any elements. You may **not** use the min function in the python library.

Examples:

find_smallest([]) should generate an error message and exit
find_smallest([1]) should return 1
find_smallest([1,2]) should return 1
find_smallest([3,1,2,0,5]) should return 0

9: [Moderate] Write a python program that reads integers from the user until the user enters 0. After the user enters 0, your program should display a list of the unique integers entered by the user without any duplicates. Do not include the 0 in your output. The order of the numbers in the output is not important.

Sample Input:

Enter a number: **7**
Enter a number: **1**
Enter a number: **7**
Enter a number: **1**
Enter a number: **10**
Enter a number: **1**
Enter a number: **0**

Sample Output:

The unique numbers entered were:
7
1
10

10: **[Difficult]** Write a Python function named merge that takes two lists of integers as parameters. Each of the lists that you are provided with will already be in sorted order. Your function should merge the two lists to form a new list in sorted order. You may not use the sort method or the sorted function from the Python library.

Examples:

merge([], []) should return []
merge([1,2,3,4], []) should return [1,2,3,4]
merge([], [1,2]) should return [1,2]
merge([2,4,5,6,9], [3,7,8]) should return [2,3,4,5,6,7,8,9]
merge([1,2,3], [1,2,3]) should return [1,1,2,2,3,3]