COSC 101, Exam #1
February 2019

Name: ⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯

Please write your name above. Do not start the exam until instructed to do so.

You have 50 minutes to complete this exam.

There are 6 questions and a total of 41 points available for this exam. Don't spend too much time on any one question.

Since indentation is important in Python, please be sure that your use of indentation is obvious for any code you write.

If you want partial credit, show as much of your work and thought process as possible.

If you run out of space for answering a question, you can continue your answer on one of the blank pages at the end of the exam. If you do so, be sure to indicate this in two places: (1) below the question, indicate which blank page contains your answer, and (2) on the blank page, indicate which question you are answering.

Question	Points	Score
1	7	
2	6	
3	5	
4	5	
5	8	
6	10	
Total:	41	

1. (7 points) Assume that the following statements have already been executed:

```
q = 'Thank you for the Valentine gift!'
y = '2019'
i = -32
f = 1.07
b = type(i) == type(round(f))
```

For each of the following expressions, evaluate the expression and write the resulting value, or identify the error in the code that would prevent it from running.

(a) `q+'  You're welcome!'`

(b) `(3 ** 2 ** 2 + -i // (int(f)+1) \% 5+1*3)/5`

(c) `y-i+2*f`

(d) `'you' not in q or b`

(e) `len(q)>len(y)>=1`

(f) `len(f)`

(g) `2*y`

2. (6 points) What is the purpose of the following program? (2 points out of 6) Draw a rectangular frame around each separate if statement in the code below (2 points out of 6). Write a code with only one chained if statement with two elif and no else to do the same thing (2 points out of 6).

```
a = float(input("Input first number: "))
b = float(input("Input second number: "))
c = float(input("Input third number: "))
if a > b:
    if a < c:
        result = a
    elif b > c:
        result = b
    else:
        result = c
else:
    if a > c:
        result = a
    elif b < c:
        result = b
    else:
        result = c

print("The result is", result)
```

3. (5 points) What is printed by the following program if the user enters the following input?
   text1: "A car with airbag"
   text2: "banana is the best"
   character: a

```python
text1 = input('text1:')
text2 = input('text2:')
character = input('What character are you looking for?')


count1=0
for char in text1:
    if char == character:
        count1=count1+1
count2=0
for char in text2:
    if char == character:
        count2=count2+1
if count1==count2:
    if count1<5:
        print (character*count2*2)
    else:
        print (character*5)
if count1%2==1:
        print (character*(1+abs(count1-count2)))
elif True:
        print ('I should always be printed')
else:
        print ("I don't think so")
```

4. (5 points) The owner of a local shop is tired of writing receipts and adding numbers by hand.
   They hire you to write a receipt program totaling scanned items. Items can either be articles
   (with a positive cost) or coupons (with a negative cost). Write a program that first asks for the
   number of items to be scanned and then asks for the cost of each item. Finally the program
   should print the total amount owed and the total savings if any from the coupons. Here is an
   example for three items, including one coupon.

   ```
   How many items are being scanned?: 3
   Cost of item 1: 4.5
   Cost of item 2: 5.3
   Cost of item 3: -1.01
   The total cost of all 3 items is $8.79.
   You saved $1.01 with your coupons!
   ```

   Here is another example with multiple coupons:

   ```
   How many items are being scanned?: 5
   Cost of item 1: 1.75
   Cost of item 2: -4.97
   Cost of item 3: 2
   Cost of item 4: -0.03
   Cost of item 5: 1.3
   The total cost of all 5 items is $0.05.
   You saved $5.0 with your coupons!
   ```

   You may assume that the total cost of all items is not negative. The first three lines of the code
   are

   ```
   num_items_to_scan = int(input("How many items are being scanned?: "))
   total_cost = 0.0
   total_coupon_cost = 0.0
   ```

   The last two lines of the code are

   ```
   print("The total cost of all", num_items_to_scan,\
   "items is $" + str(total_cost) + ".")
   print("You saved $" + str(-total_coupon_cost) + " with your coupons!")
   ```

   For the rest of the code, select one line of code from each of the pairs of lines of code given in
   the next page and reorder them to solve the problem:

5

```
A1      total_cost + item_cost
A2      total_cost += item_cost

B1      item_cost = float(input("Cost of item " + item_number + ": "))
B2      item_cost = float(input("Cost of item " + str(item_number+1) + ": "))

C1         total_coupon_cost -= item_cost
C2         total_coupon_cost += item_cost

D1  for item_number in range(num_items_to_scan):
D2  for item_number in range(num_items_to_scan+1):

E1      if item_cost < 0.0:
E2      if total_cost <= 0.0:
```

Select only 5 lines of code from above, and only one line from each pair. You may fill in line identifiers (*e.g.*, E2) below, or write out the code.

_____

_____

_____

_____

_____

5. (8 points) Write a program which will find all such numbers which are divisible by 7 but are not a multiple of 5, between 2000 and 3200 (both included). The numbers obtained should be printed in a comma-separated sequence on a single line. As an example to show what the output looks like, the following is the output of the same problem for the numbers between 7 and 30: 7, 14, 21, 28

   Note: The output must not end with a comma. Do not remove the last comma by a string method.

6. (10 points) In several sports competitions where there are multiple judges of an athlete's performance, the average score computed after an athlete competes is a *truncated average*. That is, the highest and lowest scores are dropped before then computing the average value. For example, if the judges' scores for a given event are 5.5, 8.7, 8.0, 7.3, and 9.9, the lowest (5.5) and highest (9.9) scores would be dropped and the average would be `(8.7+8.0+7.3)/3=8.0`.

Write a program that asks for the number of scores to input, then repeatedly asks for that number of scores from the user. You can assume that all scores are between 0 and 10, inclusive. You can also assume that at least one score is entered.

After all scores are entered, the truncated average should be printed. However, if fewer than 3 scores are entered, the program should print a non-truncated average and a message "not truncated".

(This page is intentionally blank. Label any work with the corresponding problem number.)

(This page is intentionally blank. Label any work with the corresponding problem number.)