COSC 101, Exam #2 Practice

Name & Section: _____

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

1. **While loop tracing**

   (a) The following function is meant to calculate and return the sum of all integers between 1 and n (inclusive), but there are lines missing from the function. As a result, the function has a runtime error (variables not defined) and a semantic error (it does not calculate the sum correctly). Write three lines of code to fix the errors (please be careful about your indentation).

```
def sum_of_n(n):



    while 0 <= num <= n:
            sum += num



    return sum
```

Solution:

```
def sum_of_n(n):
    sum = 0
    num = 1
    while 0 <= num <= n:
        sum += num
        num += 1
    return sum
```

   (b) What does this program print? If there is an infinite loop, indicate the first 4 lines of what the program prints, and state that there is an infinite loop.

```
x = 4
while x != 6:
        if x > 6:
                x = x - 2
                print(x)
        else:
                x = x + 3
                print(x)
```

Solution:

```
7
5
8
6
```

(c) What does this program print? If there is an infinite loop, indicate the first 4 lines of what the program prints, and state that there is an infinite loop.

```
x = 2
while x != 0:
    if x > 2:
        x = x - 1
        print(x)
    else:
        x = x + 2
        print(x)
```

Solution: Infinite loop. First 4 lines of output are:

```
4
3
2
4
```

2. Your instructor is looking to buy a stove for the winter and she has tasked you to write a program that reads from the input names of companies that produce stoves and prices of their items, and for each company prints the average cost of a stove.

   Complete the two function implementations defined on the other page, to produce the sample output below.

```
Type company name: GE
Enter price of stove (-1 when done): 300
Enter price of stove (-1 when done): 400
Enter price of stove (-1 when done): 250
Enter price of stove (-1 when done): -1
The average cost for GE is 316.6666666666667
Type company name: Bosch
Enter price of stove (-1 when done): 405
Enter price of stove (-1 when done): 395
Enter price of stove (-1 when done): -1
The average cost for Bosch is 400.0
Type company name: bye
```

```
def get\_average\_stoves() -> float:
    '''This function gets invoked for each company and
    reads all the item prices for this company from
    the user input (until the user enters -1) and
    returns their average.
    '''
```

```
def main() -> None:
    '''
    This function reads from the user input the name
    of each company until 'bye' is entered and calls
    get\_average\_stoves() to get the average for
    each company and then output this information
    '''

    company_name = input('Type company name:')
```

Solution:

```
def get\_average\_stoves() -> float:
    '''This function gets invoked for each company and
    reads all the item prices for this company from
    the user input (until the user enters -1) and
    returns their average.
    '''

    total = 0
    count = 0
    price = float(input('Enter price of stove (-1 when done): '))
    while price != -1:
        count += 1
        total += price
        price = float(input('Enter price of stove (-1 when done): '))

    return total / count
```

```
def main() -> None:
    '''
    This function reads from the user input the name
    of each company until 'bye' is entered and calls
    get\_average\_stoves() to get the average for
    each company and then output this information
    '''

    company\_name = input('Type company name:')

    while company\_name != 'bye':
        average\_price = get\_average\_stoves()
        print(f'The average cost for {company\_name} is {average\_price}')
        company\_name = input('Type company name:')

main()
```

3. What is the output of calling `mystery` function with values 240 (for pumpkins), 105 (for people) and 1 (for minperperson)? Draw the control flow chart. Provide different values to pass as arguments to the function in order to get different outcomes.

```
def mystery(pumpkins: int, people: int, minperperson: int) -> None:
    print(f"Out of {pumpkins} (pumpkins), {people} (people),
                                and {minperperson} (minperperson):")
    if (pumpkins < people * minperperson):
        print("We need more pumpkins!")
    elif (pumpkins >= people * minperperson):
        print("We have enough pumpkins.")
        pumpkinsperperson = pumpkins // people
        if (pumpkinsperperson >= 1):
            print("Each person may carve", pumpkinsperperson,
                                        "pumpkin(s).")
        else:
            print("At least", 2 * people - pumpkins, "people must share
                                        a pumpkin.")
        if (pumpkinsperperson == 1 and pumpkins % people == 0):
            print("There will be no extra pumpkins.")
        else:
            print("There may be extra pumpkins.")
    else:
        print("We have bigger problems than pumpkins!")
```

Solution:

```
Output with 240 (pumpkins), 105 (people), and 1 (minperperson):
We have enough pumpkins.
Each person may carve 2 pumpkin(s).
There may be extra pumpkins.

Three new values to get different output:
120 (pumpkins), 100 (people), 2 (for minperperson)
```

(This page blank for question 3 workspace.)

4. Write a function called `diff_type` that takes in a list of numbers and prints `+` when the difference between two consecutive elements of the list is positive, `=` when they are equal, and `-`, otherwise.

   For example, `diff_type([3,4,1,7])` must output `'+-+'` because the values are increasing between 3 and 4 and 1 and 7 (which warrants a '+'), and decreasing between 4 and 1 in the middle (which warrants a '-');`diff_type([12,30,30])` must output `'+='`, `diff_type([32,16,8,4,2,1,0])` must output `'------'`

Solution:

```
def diff_type(alist: list) -> None:
    res = ""
    for i in range(len(alist)-1):
        num = alist[i]
        next = alist[i+1]
        diff = next - num
        if diff > 0:
            res += '+'
        elif diff < 0:
            res += '-'
        else:
            res += '='

    print(res)
```