Name:_____ Section:_____ Andrew Id: _____

15-110 Spring 2019 Quiz6

* 30 minutes, No calculators, no notes, no books, no computers, no phones

1. **Code Tracing [5 pts]** Indicate what the following program prints. Place your answer in the box.

```
def ct1(n):
    if (n < 10):
        return n
    else:
        return (n%10)**2 + ct1(n//10)
print(ct1(10231))
```

2. **Code Tracing [5 pts]** Indicate what the following program prints. Place your answer in the box.

```
def ct2(s):
    if (s == ''):
        return s
    else:
        c = s[-1]
        if (c.isalpha() == True):
            c = ''
        return c + ct2(s[0:-1])
print(ct2('a1-B2?'))
```

3. LMC Code Tracing [10 pts]

What will the following LMC code output when run on an input of 4? Circle your answer.

Hint: BRP means "Branch if the Accumulator is Positive or Zero (not Negative)"

```
        INP
        STA COUNTER
LOOP    LDA RESULT
        ADD COUNTER
        STA RESULT
        LDA COUNTER
        SUB DELTA
        STA COUNTER
        BRP LOOP
        LDA RESULT
        OUT
        HLT
COUNTER DAT 000
DELTA   DAT 002
RESULT  DAT 000
```

4. **Short Answers [30 pts; 5 pts each value]**

1. Which best describes top-down program design?

A) Write some low-level helper functions, then piece these together into higher-level functions, and keep doing this until you solve your problem.

B) Write the top-level solution assuming you have any helper functions you need, and only then write those helper functions (assuming any helper functions they in turn need), and so on, until you solve your problem.

C) Write a top-level description of the problem, in English, and only once you understand that well, proceed to write the Python code.

D) Write your code in a Python file starting at the top of the file, and proceeding down the file until you solve your problem.

2. According the source cited in the notes, about how much per year do software errors cost the global economy?

A) $1.7 million

B) $1.7 billion

C) $1.7 trillion

D) $1.7 quadrillion

E) $1.7 quintillion

3. Circle ALL of the following that are real style rules according to our style guide:

A) Use short variable names to save typing

B) Use comments to make code clearer

C) Use large functions to keep your logical blocks of code together.

D) Avoid global variables as they can lead to complicated bugs.

Note: answer each of the following questions VERY BRIEFLY in just a few words. Really, do not use sentences let alone paragraphs.

4. What was the main goal of Turing's work at Bletchley Park (that is, why did he build Christopher)? Be brief but specific.

5. Who in Turing's life was the real Christopher (the person after whom Turing named his famous computer)?

6. What was so important about the phrase "Heil Hitler" that it ultimately helped lead to the success of Turing's team? Again, be very brief but specific.

5. **Free Response: sumOfAllDigits(L) [50 pts]**

Without using any loops, write the recursive function sumOfAllDigits(L) that takes a possibly-empty list L of integers and returns the sum of all the integers in all of L. For example:

```
assert(sumOfAllDigits([12, 23]) == 8) # 1+2+2+3
assert(sumOfAllDigits([12, 203, 2]) == 10) # 1+2+2+0+3+2
assert(sumOfAllDigits([ ]) == 0) # no digits, so the sum is 0
```

Hint: you may write a helper function here, say digitSum(n) that sums the all the digits in one number, so long as it is also recursive (and does not use any loops).

6. **Bonus LMC Code Tracing [2.5 pts]**

What will the following LMC code output when run on an input of 8?   Circle your answer.

```
        INP
        STA COUNTER
LOOP    LDA RESULT
        ADD RESULT
        STA RESULT
        LDA COUNTER
        SUB DELTA
        STA COUNTER
        BRP LOOP
        LDA RESULT
        OUT
        HLT
COUNTER DAT 000
DELTA   DAT 001
RESULT  DAT 001
```

7. **Bonus Code Tracing [2.5 pts]** Indicate what the following program prints. Place your answer in the box.

```
def bonusCt(L):
    if (L == [ ]):
        return L
    else:
        i = len(L)//2
        return (bonusCt(L[:i]) + [i] +
                bonusCt(L[i+1:]))
print(bonusCt(list(range(8))))
```