CS 115 Exam 2 Thu. 10/27/2016

Name: ______________________________

Rules and Hints

- You may use one handwritten 8.5 × 11" cheat sheet (front and back). This is the only additional resource you may consult during this exam. *No calculators.*
- Explain/show work if you want to receive partial credit for wrong answers.
- As long as your code is correct, you will get full credit. No points for style.
- When you write code, be sure that the indentation level of each statement is clear.

Grade

Problem	**Your Score**	**Max Score**
Problem 1: Trace snippets of code		20
Problem 2: Trace functions		20
Problem 3: Define functions		30
Problem 4: Write a complete program		30
Total		100

Problem 1: Trace snippets of code (20 points)

What will print to the screen when each of the following snippets of code is executed in PyCharm or in the Online Python Tutor?

Be very clear with spacing, line breaks, etc.

Treat each sub-problem as an independent question.

Problem 1A

```
x=4
y=10
while x < y:
  x=x+2
  print(x, end=" ")
```

Problem 1B

```
s = 'President of USA'
print(len(s))
print(s[len(s)-1])
print(s[10:15])
print(s[:7])
```

Problem 1C

```
strings = ['Election', 'Day', 'Is', 'November', '08']
print(len(strings))
print(strings[4])
print(len(strings[1]))
print(strings[1][2])
```

Problem 1D

```
A = [2, 4, 3]
B = A
C = A[:]
B[0] = 5
C[2] = 5
print(B)
print(A+C)
```

Problem 1E

```
value = '3.17'
try:
    pi = int(value)
    print('pi = ', pi)
except Exception as e:
    print('pi = ', value)
```

Problem 2: Trace functions (20 points)

What will print to the screen when each of the following snippets of code is executed in PyCharm?

Be very clear with spacing, line breaks, etc.

Note: the parts of this problem are independent.

Problem 2A

```
def calculate(x, y):
    return x - y


z = calculate(1, 2)
print(z)
```

Problem 2B

```
def mystery(str):
    new_str = ''
    for char in str:
        new_str = char + new_str
    return new_str


newStr = mystery("disney")
print(newStr)
```

Problem 2C

```
def getFirstElem(A):
    return A[1]

A = [6, 3, 4, 0, 5, 3]
print(getFirstElem(A))

A = [[6, 3], [4, 0], [5, 3]]
print(getFirstElem(A))
```

Problem 2D

```
def update_list(list):
    for elem in list:
        elem = elem.lower()
    print(list)

    for i in range(len(list)):
        list[i] = list[i].upper()
    print(list)


list = ["Celebrating", "Democracy"]
update_list(list)
print(list)
```

Problem 3: Define functions (30 points)

Define functions to perform the following tasks, obeying these requirements:

- Assume that the math library has been imported for you.
- Do NOT ask the user for input unless the specification explicitly requires it.
- Do NOT print anything unless the specification explicitly requires it.

Problem 3A

Define a function named `cyl_vol` that:

- Takes 2 parameters: the radius and height of a cylinder
- Returns the volume of the cylinder, which is given by

$$\text{Volume} = \pi r^2 h$$

Problem 3B

Define a function named `remove_spaces` that:

- Takes 1 parameter: a string
- Returns a string with spaces removed

Example of calling this function: `remove_spaces("a b c")` will return `"abc"`

Problem 3C

Define a function named `maxColumnSum` that:

- Takes 1 parameter: a 2D list of integers (All sublists are of the same length).
- Computes sum of each column.
- Returns the maximum of these sums.

For example, `maxColumnSum( [ [15, 10, 5], [1, 12, 3] ] )` will return `22`

Problem 4: Write a complete program (30 points)

For this problem, you must write a complete program. That includes a docstring, a `def main()`, any necessary library imports, etc.

To get full credit, your functions should call each other where appropriate to avoid duplicating work!

Your program should contain the following functions:

- `getCandidates`: Takes no parameters. Prompts the user for names of presidential candidates until the user enters "go". Returns a list of the names the user entered (not including the word "go"). Each line of user input should count as a single name, even if it has multiple words.

  For example:

  ```
  Enter a candidate's name:  Jeb Bush
  Enter a candidate's name:  Hillary Clinton
  Enter a candidate's name:  Donald Trump
  Enter a candidate's name:  go
  ```

  should return the list `[ 'Jeb Bush', 'Hillary Clinton', 'Donald Trump']`.

- `num_vowels`: Takes a string as a parameter and returns the number of vowels in it (*lower case or upper case*):

  For example:
  `num_vowels('Mitt Romney')` is 3
  `num_vowels('Barack Obama')` is 5
  `num_vowels('Jeb Bush')` is 2

- `main`: Calling your other functions whenever possible, this function should get a list of names from the user (stopping when the user enters "go"). It should then determine the winner based on the name that has the *least* number of vowels. The function should print the following:
  - Each name followed by number of vowels in it.
  - The name of the winner in upper case.

Problem 4 code

Problem 4, continued