NAME: _____
Please sign the Honor Pledge on the last page.

CSCI 150
Exam 2
April 22, 2016

There are 6 numbered questions here. Numbers 1-4 are worth 15 points each. The two parts of question 5 are worth 10 points each. Question 6 is worth 20 points.

1. What will this print? Read it carefully.

```
def foo(D, key, value):
    D[key] = value

def bar(D):
    for k in D.keys():
        print( k, D[k])

def main():
    D = {}  # keys is a string; value is a number.
    foo(D, "A", 23)
    foo(D, "B", 42)
    foo(D, "A", 57)
    foo(D, "C", 15)
    bar(D)

main()
```

Answer:

A 57
B 42
C 15
in any order

2. What will this print? Read it carefully.

```
class What:
        def __init__(self, name):
            self.name = name
            self.value = 0

        def foo(self, x):
            self.value = x

        def __str__(self):
            return self.name

def main():
    D = What(  "bob" )
    D.foo(23)
    D.foo(45)
    D.foo(71)
    print(D)

main()
```

This prints “bob”/

3. What is wrong here? The following program makes a list L1 that contains some prime numbers, and it tries to copy that data into a new list L. This crashes with an error message that says "List assignment index out of range" for the line B[i]=A[i]. Explain what is wrong in one English sentence and correct the code so that the Copy f unction will work.

```
def Copy(A, B):
        # this copies list A to list B
    for i in range(0, len(A)):
        B[i] = A[i]

def main():
    L1 = [2, 3, 5, 7, 11, 13, 17]
    L = []
    Copy(L1, L)
    print(L)

main()
```

When we call Copy(L1, L) the second argument is empty, so
B[i]=A[i]
makes no sense; there is no B[i]

Possible fixes inclde

```
def Copy(A, B):
        for i in range(0, len(A)):
            B.append(A[i[)
```

or

```
def Copy(A, B):
        for elt in A:
            B.append(elt)
```

or

```
def Copy(A, B):
        B.extend(A)
```

4. Here is a Person class and a short main() program that uses it. Unfortunately this crashes with the error message: "__init__ takes 1 positional argument but 2 were given."
    a) What two arguments were given to __init__?
    b) Fix the class definition so that all of the methods are correct.

```
class Person:
    def __init__(name):
        name = name
        age = 0

    def setAge(a):
        age = a

    def __str__():
        return name

def main():
    x = Person( "bob" )
    x.setAge(64)
    print(x)

main()
```

```
class Person:
    def __init__(self, name):
        self.name = name
        self.age = 0

    def setAge(self, a):
        self.age = a

    def __str__(self):
        return self.name
```

Answers:

a) "bob" and the object being constructed, which you might call self. You could also say the arguments are self and name.

b) The corrected code is in the box above and to the right.

5. a) I have a list of pairs, such as [(“bob”, 64),(“Mary, 18), (“Isabel”, 20), (“Gabe”, 24)]. Write a function FindAge(L, name) where L is such a list. This returns either the age of the person in the list with that name (such as 18 for “Mary”), or None if there is no person in the list with that name.

```
def FindAge(L, name):
    for (person, age) in L:
        if person == name:
            return age
    return None
```

or

```
def  FindAge(L, name):
    for i in range(0, len(L)):
        if L[i][0] == name:
            return L[i]][1]
    return None
```

c) Now I have a dictionary D with the data from (a): D[“bob”]=64, D[“Mary”]=18, and so forth. Write a function FindAge(D, name) where D is such a dictionary. This returns either the age of the person in the dictionary with the given name or else None if there is no such person.

```
def findAge(D, name):
    is f name in D.keys):
        return D[name]
    else:
        return None
```

6. Write a class to represent a player in a game where players take turns rolling a pair of dice (each die has the numbers 1 through 6). The class has 3 methods:
   a) The constructor takes as an argument the player's name and saves this in a variable.
   b) The Turn method takes no arguments. It rolls each of the two dice (with random.randint(1, 6)) and prints the two dice values. If the two values are the same it rolls again and the turn continues until the dice values are different. The value of each roll is added onto the player's score.
   c) The Print() method also takes no arguments. It prints the player's name and score

   For example, if we write an application program like t his:

```
def main():
    b = Player( "bob" )
    b.Turn()
    b.Turn()
    b.Print()

main()
```

   we might get output:

```
bob rolled 5 and 6
bob rolled 6 and 6
bob rolled 5 and 1
bob has 29 points
```

   Your job is to write class Player.

```
class Player:
    def __init__(self, name):
        self.name = name
        self.score = 0

    def Turn(self):
        done =False
        while not done:
            a = random.randint(1, 6)
            b = random.randint(1, 6)
            self.score = self.score+a+b
            print( "%s rolled %d and %d"%(self.name, a, b))
            if a != b:
                done = True
    def Print(self)L
        print( "%s has %d points."%(self.name, self.score))
```