

COSC 235 Spring 2007
Quiz #3

Please read each question carefully and be sure to give complete answers. Good luck!

1. (1 pt.) What is your name? _____

2. (8 pts.) Complete the following truth table:

p	q	r	$p \text{ and } q$	$\text{not } r$	$(p \text{ and } q) \text{ or } (\text{not } r)$
T	T	T			
T	T	F			
T	F	T			
T	F	F			
F	T	T			
F	T	F			
F	F	T			
F	F	F			

3. (12 pts.) Write a *while*-loop to do each of the following:

a) Find the sum of the first n counting numbers: $1 + 2 + 3 + \dots + n$

b) Find the sum of the first n even numbers: $2 + 4 + 6 + \dots + 2n$

c) Count how many times 5 can be subtracted from a number x before the number is negative.

4. (8 pts) Rewrite the following pre-test loop using Python's "post-test" loop idiom.

```
cash = -1
while cash < 0:
    cash = input("Enter a positive number: ")
    if cash < 0:
        print "Try again"
```

5. (16 pts.) Write two functions `die()` and `dice()` that model a pair of dice. `die()` should generate a random number, from 1 to 6 inclusive for the value of a roll of a single die. `dice()` should use `die()` to generate a pair of random number corresponding to the roll of a pair of dice. It should return the values generated as a pair and should display the values, their sum, and, if appropriate a message "You rolled doubles!". For example:

```
>> d1, d2 = dice()
Your rolled doubles. You rolled 6 and 6, a total of 12.
>> d1, d2 = dice()
You rolled 3 and 5, a total of 8.
```

Note: This function both returns and displays values!

6. (12 pts.) Given arbitrary lists `lst1` and `lst2`, give a command to do each of the following:

- a) find `x` such that `lst1[x]` returns "sam" where "sam" is some item on the list.
- b) adds "sue" to the end of `lst1`.
- c) adds "sue" to `lst1` so that "sue" is the second item on `lst1`.
- d) create a new list `lst3` that has everything `lst1` and everything in `lst2`.
- e) removes and returns the last item in `lst1`.
- f) returns the number of times "paul" is in `lst1`.

7. (12 pts.) Using the dictionary

```
d = {"jim":92, "jimbo":85, "james":99, "jimmy":71}
```

tell what is returned or explain why an error is generated by each of the following:

- a) `d[jimbo]`
- b) `d.keys()`
- c) `d.has_key("jimmie")`
- d) `d.get("waldo", 100)`
- e) `85 in d`
- f) `d.items()`

8. (8 pts.) The Fibonacci sequence starts 1, 1, 2, 3, 5, 8, That is, each number is the sum of the two previous numbers. Thus, the next number is $5 + 8$ or 13. Write a function that returns the n^{th} Fibonacci number. For example, `fib(6)` would return 8.

The code on the next page is an incomplete class definition for a class `Tests` that records the test grades for a course. Use this code to answer the following questions.

9. (3 pts.) In the appropriate space, add a *docstring* of your choosing to the class.
10. (8 pts.) In the appropriate space, write a method `average` that calculates and returns the grade average for the tests taken in a course. Your code should return 0 in no tests have been recorded and the arithmetic mean otherwise.
11. (12 pts) In the space below the code, give code to
- a) create two instances of `tests`, `COSC235` and `MATH235`
 - b) recorded the grades 98, 92, and 96 for `COSC235`
 - c) recording the grades 90 and 92 for `MATH235`
 - d) display the second recorded grade for `COSC235`
 - e) display the average for each course
 - f) display the highest grade recorded for `MATH235`

```
## Attached is the code for the preceding problems

## class definition
class tests:

    ##constructor
    def __init__(self):
        self.grades = []

    ## record a grade
    def record(self, score):
        self.grades.append(score)

    ## return a grade based on it position in the list
    def report(self, number):
        return self.grades[number - 1]

    ## return the current average grade

    ## utilitiy function
    def listMax(self, lst):
        x = lst[0]
        for i in lst:
            if x < i: x = i
        return x

    ## find the highest grade
    def maxGrade(self):
        return self.listMax(self.grades)
```

Pledged: _____