About Exam 1

*There are three types of questions I can ask. There will be at least one question of each type.*

Type I. Here is some code; what does it do? Typically the program will print something and I’ll want to know what it prints.

Type II. Here is some buggy code and an error message (or some incorrect output). Explain what is wrong and how to fix it.

Type III. Write a function (or a program) to do X.

**Topics:**

A. Basic types: integers, booleans and strings  
B. Variables and variable names  
C. Basic statements  
  - Assignment statements  
  - `input()` statements  
  - IF-statements  
  - WHILE-loops  
  - FOR-loops  
D. Functions:  
  - Function definitions  
  - Arguments and Parameters  
  - Function calls  
  - Functions returning values  
E. Structures  
  - Files (opening and reading with a for-loop)  
  - Lists  
F. Other  
  - Exceptions and try-catch statements  
  - Random numbers

By far the most important item on this list is FUNCTIONS. LOOPS and LISTS take second and third places.
Here are some typical questions:

1. The following program is supposed to ask the user for a number and say if it is prime. Unfortunately it says that every odd number is prime. How can we fix this?

   ```python
def isPrime(num):
    for x in range(2, num):
        if num % x == 0:
            return False
    else
        return True

def main():
    x = input("Gimme a number: ")
    if isPrime(x):
        print("%d is prime." %x)
main()
```

2. What is wrong with the following program? When I run it I get an error message

   ```python
   def Average(L):
    sum = 0.0
    for x in L:
        sum = sum + x
    average = sum/len(L)

def main():
    average = Average([2, 6, 4, 1, 7])
    print("The average is %.2f" % average)
main()
```

print "The average is %.2f" % average
TypeError: float argument required, not NoneType

Explain (1 sentence is enough) what this error message means and say how to fix the problem.
3. What will this program print?

```python
def Test(s):
    # Variable s will be a string
    for x in s:
        if x == "b":
            return True
    return False

def Foobar(L):
    # Variable L will be a list of strings
    for x in L:
        if Test(x):
            print("%s: yep" % x)
        else:
            print("%s: nope" % x)

def main():
    Foobar(["Marvin Krislov", "Oberlin College", "B Geitz"])

main()
```

4. Write a program that asks the user for a number n, then prints n random numbers between 1 and 6 (like rolls of a dice) and ends by printing their sum. Here is a typical run:

```
How many rolls? 3
You rolled 5
You rolled 3
You rolled 5
The sum of your rolls is 13
```
5. Write a function `censor(WordList, BadWords)` that takes a list of words and a list of forbidden words. It returns a new list with the entries of WordList, but any word in the BadWord list is replaced by “XXX”. So if WordList is
[“Today”, “is”, “exam”, “day”, “yippee”, “I”, “love”, “a”, “good”, “exam”]
and BadWords is [“test”, “exam”, “grade”, “joy”, “good”, “yippee”] then
Censor(WordList, BadWords) returns
[“Today”, “is”, “XXX”, “day”, “XXX”, “I”, “love”, “an”, “XXX”]

6. Write a function `fewerVowels(s, t)` that takes two strings s and t and returns the one with fewer vowels. If s and t have the same number of vowels you can return either. For example, FewerVowels(“bob”, “marvin”) returns “bob”. You can take A, E, I, O, U as the only vowels, but do consider both upper – and lower- cases.

7. We could say that a list of numbers is *increasing* if each element is larger than the previous one. The list [1, 3, 4, 10] is increasing but the list [4, 2, 5] is not. Write a function `isIncreasing(L)` that returns True if list L is increasing and False if it is not.