Exam 1 Material

1. (40 points) If A = 4 and B = 7, show what each of the following statements, or pair of statements produces. Indicate a space by an underscore (so _ _ represents two spaces).

a. print("A = ", A, B)

b. print("A=", A, end = "\$")
print("B=", B)

c. print("Result = ", A*B)
print(end="\n")
print('No more')

d. print("A + str(A), sep = "\?")
2. (24 points) Describe the following stages of the programming process. You may use examples to help answer the questions.

a. Problem description/analysis

b. Test/debug

c. Maintenance

3. (12 points) Are the following valid python variable names?

a. %x

b. x yz

c. x8j

d. 8yes

e. eight

f. _100test
4. (14 points) What are the data types (int or str) of the following expressions:

a. 'XYZ'

b. “23”

c. “18.3”

d. eval("23")

e. 23

f. eval("3 ** 2")

g. 3 – 2 * 3

5. (10 points) Write a program that prompts for your name (name), current year (year), and date of birth (dob). Print out name, year, dob, and year-dob.

Exam 2 Material

1. (18 points) In the blank to the right, write the value that Python would calculate. If an error would be produced, write the word error and circle the place in the expression that the error arises. If a value is a string you must show quote marks around the string.

a. 
   x = 5 – 4 - 3
   
   b. 
   x = 2 ** 3 ** 2
   
   c. 
   x = 7 * 2 ** 3 + 9 / 2
   
   d. 
   x = 11 / int(2.5) // 2
   
   e. 
   x = 8 - 5 / (9 - 4)
   
   f. 

2. (20 points) What is the output of each of the following code segments? Indicate spaces as appropriate using an underscore character in the previous problem.

a. for i in range(3):
   print (i**2, end = " ")

b. for i in range(5,8):
   print (i // 2, end = " ")

c. for i in range(18, 6, -3):
   print (i)

d. total = 0
   for i in range(1, 11, 2):
      total = total + 1
   print (total)

3. For the following binary/decimal conversions assume the leftmost bit is the sign bit.
   a. Represent the decimal number -55 in a byte.

   __________________________

   b. Represent the binary number 01101100 as a base 10 integer.

   __________________________

4. An integer is stored in 32 bits. Using base 2 notation what is the range of integer values?
   ______________________________________________________
I'm only going to ask you two such problems on the actual exam, but I've put together several for practice:

2. Write a code segment that calculates the product of a set of n values input by the user.

3. Write a program to ask the user for the values of h and g. Then calculate the following and output the results.

3. (10 points) Write a code segment that calculates the average of n numbers.
Remember -- n! = n*(n-1)*(n-2) ... 1

4. (20 points) Write a code segment that finds the minimum of n numbers entered.

5. (20 points) Write a code segment that outputs n numbers. Each number is the multiplication of the previous two numbers. You can assume the first two numbers are 1 and 2, and that n > 2
Example: 1,2,2,4,8,32

6. (10 points) Write a code segment that prompts the user for n and then calculates the n!. 
Remember \( n! = n*(n-1)*(n-2) \ldots 1 \)

8. Write a program that asks the user for \( n \), then adds \( n \) terms of the series:

and displays the result.

Exam 3 Material

1. Write a code segment that creates a window that is 300 pixels wide and 400 pixels tall and then draws a rectangle with one corner at location (50, 30) and its diagonally opposite at location (150, 200). The user should then be prompted (from the Python shell) to strike any key to close the window. Assume the graphics package has been imported with the directive from graphics import *
2. Append to the following segment of code additional lines that:
   - Will prompt the user (in the shell) for the number of mouse clicks to expect from the user. Assume it will be an integer greater than 1.
   - Upon receiving the first click from the user the program will draw a point at the location where the user clicked.
   - Upon receiving additional clicks from the user the program will draw a line from the location where the user clicked previously clicked to the latest location where the user clicked. This continues until the number of mouse clicks specified in the user input have been received. When the drawing is done the user should see a collection of connected line segments in the window.
   - Upon receiving one additional mouse click the window will close.

   from graphics import *
   from math import *
   def main():
       win = GraphWin("CSCI220",400,200)

3. Write a program that computes the fuel efficiency of a multi-leg journey. The program will first prompt for the starting odometer reading and then get information about a series of legs. For each leg, the user enters the current odometer reading and the amount of gas used (separated by a space). You can assume that there will be n legs (supplied by the user). The program should print out the miles per gallon achieved on each leg and the total MPG for the trip AFTER all of the information has been entered.
4. Write a program that accepts a string representing a person’s first, middle and last name and returns their corresponding Edisto email address all in lower case. For example, Jane Elizabeth Monferdini should return “jemonferdi@edisto.cofc.edu”. The email address is the first initial of the person’s first name, the first initial of the middle name, and the first 8 characters of the last name followed by the at sign and “edisto.cofc.edu”.

5. Assuming the assignment statement:

```python
text = “Sue, Bill, and Bob”
```

What is the value of x in each of the following?

a. `x = text.find(“,“)`  
   x = _____________________

b. `x = text.split(“,“)`  
   x = _____________________

c. `y = text.split(“ “)`  
   `x = len(y[0])`  
   x = _____________________

d. `x = chr(ord(text[0]))`  
   x = _____________________

e. `x = text[0:4]`  
   x = _____________________
f. \[ x = \text{text}[5] + \text{text}[2] \times 2 \]

\[ x = \text{______________________} \]

6. Show the string that would result from:

(a) “Looks like {1} and {0} for breakfast”.format(“eggs”, “spam”)

(b) “Hello {1}”.format(“Susan”, “Computewell”)

(c) “{1:0.2f} {0:0.2f}”.format(2.3, 2.3468)

(d) “[{1:3}]”.format(“14”)

**Exam 4 Material**

1. Write a segment of code that opens the file Test2-3.in and determines the average number of words per line in the text.

2. Write a segment of code that opens the file Test2-4.in and copies every odd numbered line to Test2-4.out.
3. Write a conditional statement that prints the appropriate message based on user input. Assume the user input is stored in a variable `num` and is an integer.

<table>
<thead>
<tr>
<th>User Input</th>
<th>Appropriate Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 or less</td>
<td>&quot;Error&quot;</td>
</tr>
<tr>
<td>1..10</td>
<td>&quot;Good&quot;</td>
</tr>
<tr>
<td>11..15</td>
<td>&quot;OK&quot;</td>
</tr>
<tr>
<td>16 or more</td>
<td>&quot;Awful&quot;</td>
</tr>
</tbody>
</table>

6. Write a function `changeName(name)` that takes a string value `name` as an argument, where `name` is in the form

"Last, First Middle"

for example "Anderson, Paul Anderson". Note the use of the comma. The function should return a string that has the name in the form

"First Middle Last"

for example "Paul Edward Anderson."

7. Write a function `changeName2(nameList)` that takes a list `nameList` as an argument that
contains a person’s name, where name is in the form [Last, First, Middle]

for example ['Anderson', 'Paul', 'Edward']. The function should not return a value, but instead should modify the list so the name is in the form [First, Middle, Last]

for example ['Anderson', 'Paul', 'Edward'].

8. Write a function findValidValue() that returns a valid value entered by the user. The function should ask the user to enter a number until the user enters a value that is any of the following.
a. Inclusively within the range of 5 and 17.
b. Exactly equivalent to 100.
c. Exclusively within the range of 20 and 25.

8. Evaluate the following Boolean expressions:

-3 < 7
3 > 20 and 30 >= 20
20 == 20
7 == 8 and not(20 < 18)
3 > 4 or (5 <= -3 and 8 == 7)
Exam 5 Material
1. Write a function findValidValue() that returns a valid value entered by the user. The function should ask the user to enter a number until the user enters a value that is any of the following.
   a. Inclusively within the range of 5 and 17.
   b. Exactly equivalent to 100.
   c. Exclusively within the range of 20 and 25.

```python
def findValidValue():
    while True:
        value = eval(input("Enter a number: "))
        if value >= 5 and value <= 17:
            return value
        if value == 100:
            return value
        if value >= 20 and value <= 25:
            return value
```

2. Write a program that computes the fuel efficiency of a multi-leg journey. The program will first prompt for the starting odometer reading and then get information about a series of legs. For each leg, the user enters the current odometer reading and the amount of gas used (separated by a space). The user signals the end of the trip with a blank line. The program should print out the miles per gallon achieved on each leg and the total MPG for the trip.

```python
last_odometer = eval(input("Odometer reading: "))
sum = 0
legs = 0
while True:
in = input("Enter odometer reading and amount of gas (blank line signals end of trip)"
if in.strip() == ":
    break
fields = in.split(" ")
odometer = eval(fields[0])
gas = eval(fields[1])
print((last_odometer - odometer)/gas)
sum = sum + (last_odometer - odometer)/gas
legs = legs + 1
last_odometer = odometer
print(sum/legs)
```

11. Using an indefinite loop (i.e., while), write a program that asks the user for a list of numbers that is not known at the start of the program. The user will enter a blank line when they are done entering numbers. Calculate the average of all of the numbers entered by the user.
Also, during the execution of the while loop, keep track of the maximum and minimum value seen. You may NOT use the built-in Python max and min functions!

20. Show me an implementation of one of the following sorting functions: insertion, bubble, or selection (your choice).

21. Describe the procedure behind bubble sort.
22. Describe the procedure behind selection sort.

23. Describe the procedure behind insertion sort.

24. Show me an implementation of linear search.

25. Show me an implementation of binary search.

**Begin Questions over new material (classes)**

18. Define a class, Student, to represent a student at C of C.
   a. The data to maintain for a student is their name and grades. Create a construction which accepts the student’s name and initializes a list of grades to [].
   b. Write accessor methods for the attributes (get functions).
   c. Write a mutator method for the name (set functions).
   d. Write a method add_grade that accepts a float and adds a new grade to the list of grades. The method should not return a value.
   e. Write a method average_grade() that returns the average grade for the student.
f. Write a method to represent the student as a string (name and average grade). Make sure that your average grade is properly formatted to have only two digits after the decimal.

19. Write a program that does basic testing of the methods from your Student class.

20. Describe the motivation behind using classes with specific examples (no code required).

21. Describe how you would go about creating a poker game using classes. Tell me which classes you would use and how they would communicate in general.