

The Second Midterm Will Cover Content From:

Material covered in Lectures #1-12 (up to and including 'Functions and Function Scope')
Chapters 1, 2, and 3 (except 3.7), 5, 7 (except 7.7), and 4 of Python for Everybody
Practice Problems PP1, PP2, and PP3, PP4, PP5

Important topics to ensure you understand include: functions, loops, complex branching and logical operations, as well as applying problem solving and computational thinking to these sorts of problems.

Any questions regarding the midterm content, structure, practice questions, etc., should be posted in the Midterm and Exam Discussion board on cuLearn or on the course Discord server.

1. Write a Python guessing game. Generate a random number between 1-100 and have the user guess numbers until they are correct. Any time the user guesses wrong, the program should output a hint indicating 'guess higher' or 'guess lower'. When the user guesses correctly, print out a message indicating that they have won the game. You should put your code for this problem in the file `problem1.py`.
2. The files *studentgrades1.txt*, *studentgrades2.txt*, and *studentgrades3.txt* contain a list of students and their grades. The first line of the file represents a student's student number, while the second line represents that same student's grade. This pattern continues for the remainder of the file. You can assume any filename your functions are given will represent a file with this structure. Write a function called `getAverageGrade(filename)` which accepts a filename input argument. The function should return the average grade calculated over all students within the specified file. Write another function called `getBestStudent(filename)`. This function should return the student number of the student with the highest grade in the class. Both of your functions should be defined in the `problem2.py` file. The expected values for each file are given below:

 studentgrades1.txt: 44.28 average, 1109170 best student
 studentgrades2.txt: 49.92 average, 1178897 best student
 studentgrades3.txt: 59.28 average, 1393808 best student
3. Write a function called `analyze(filename)`, which accepts a single string argument representing the name of the file. You can assume the file name you are given will represent a file containing only integer values. The function should return the length of the longest consecutive sequence of even numbers inside the given file.

That is, if the numbers in the file were 2 4 1 6 8 3 **4 8 2** 1 3 6 5, the function should return 3 (the bolded sequence is 3 even numbers in a row).

4. Write a function called `gcd(int, int)` which returns the greatest common divisor of the two integer input values. The greatest common divisor is the largest number that divides evenly into both the given numbers. For example, the greatest common divisor for 20 and 24 is 4. If either of the input arguments are less than 1, the function should return -1. Your function should be defined inside the `problem4.py` file.