### Greatest Path Sum in Triangle (due 26 Oct 2020)

This assignment is a variation of a problem from Project Euler. You are required to find the greatest path sum starting at the top of the triangle and moving only to *adjacent* numbers on the row below.

3

7 4

2 4 6

8 5 9 3

In the above triangle, the maximum path sum is 3 + 7 + 4 + 9 = 23.

You will read your input from the file [triangle.in](https://www.cs.utexas.edu/users/mitra/csFall2020/cs313/assgn/triangle.in). The first line indicates *n* the number of rows in the triangle. This will be followed by *n* lines of data. Each line of data will have only positive integers greater than 0. The first line of data in the triangle will have one number, the second line in the triangle will have two numbers and so on. The nth line will have *n* integers.

You will apply four different approaches to problem solving to this single problem - exhaustive search (recursive), greedy, divide and conquer (recursive), and dynamic programming. The functions - brute\_force(), greedy(), divide\_conquer(), and dynamic\_prog() are wrapper functions. You cannot change the header but you can always add more helper functions.

You will have these lines of output:

The greatest path sum through exhaustive search is

23

The time taken for exhaustive search in seconds is

x

The greatest path sum through greedy search is

23

The time taken for greedy approach in seconds is

x

The greatest path sum through recursive search is

23

The time taken for recursive search in seconds is

x

The greatest path sum through dynamic programming is

23

The time taken for dynamic programming in seconds is

x

Most likely the maximum path that you will get from exhaustive search, recursive search, and dynamic programming will be the same. Whereas the result from the greedy search will be less and so will be the execution time.

Here is the template of the file called [Triangle.py](https://www.cs.utexas.edu/users/mitra/csFall2020/cs313/assgn/Triangle.py) that you will be turning in. We are looking for clean and structured design using the [standard coding conventions](https://www.python.org/dev/peps/pep-0008/)in Python. You may not change the function signatures but you may add as many helper functions as needed. The file will have a header of the following form:

# File: Triangle.py

# Description:

# Student's Name:

# Student's UT EID:

# Partner's Name:

# Partner's UT EID:

# Course Name: CS 313E

# Unique Number:

# Date Created:

# Date Last Modified:

To run this code on the command line on the Mac you will do

python3 Triangle.py < triangle.in

To run the code on the command line on a Windows machine you will do

python Triangle.py < triangle.in

For this assignment you may work with a partner. Both of you must read the paper on [Pair Programming](https://www.cs.utexas.edu/users/mitra/csFall2020/cs313/assgn/PairProg-CACM-1999.pdf) and abide by the ground rules as stated in that paper. If you are working with a partner you will be submitting only one program but make sure that you have your partner's name and eid in your program. If you are working alone, then remove the two lines that has the partner's name and eid in the header.

Use the [Canvas](http://canvas.utexas.edu/)system to submit your **Triangle.py** file. We should receive your work by 11 PM on Monday, 26 Oct 2020. There will be substantial penalties if you do not adhere to the guidelines. Remember Python is case sensitive. The name of your file must match exactly what we have specified.

* Your Python program should have the proper header.
* Your code must run before submission.
* You should be submitting your file through the web based *Canvas* program. We will not accept files e-mailed to us.