

Assignment 5 of 6**Due: Monday, November 23th by 11:30pm**

For each question in this assignment **you will be submitting three files containing source code written in Python 3, that have been compressed into a "zip" file. The Python sources (i.e., the .py files) should be named "a5q1.py" and "a5q2.py". The zip file should be named a5.zip. You will submit your file using cuLearn.**

A LATE POLICY IS IN EFFECT FOR THIS ASSIGNMENT**LATE ASSIGNMENTS WILL BE ACCEPTED FOR 48 HOURS AFTER THE DEADLINE AT A PENALTY OF 2.0% / HOUR**

You are expected to **demonstrate good programming practices at all times** (e.g., choosing descriptive variable names, provide comments in your code, etc.) and **your code may be penalized if it is poorly written**. You are also expected to **do the necessary preparatory work** (i.e., devising an algorithm) **before you start coding**.

PLEASE NOTE: YOU WILL BE ASKED TO PRESENT EITHER PSEUDOCODE OR A FLOWCHART BEFORE YOU WILL RECEIVE ANY ASSISTANCE FROM THE INSTRUCTOR OR A TEACHING ASSISTANT

Question 1 – Pascal's Triangle

For this question you will write a program that will use loops to print out rows of Pascal's triangle while looking for a specific integer value provided by the user. (Pascal's triangle is a triangular table of coefficients for the expanded binomial formula.)

1				
1	1			
1	2	1		
1	3	3	1	
1	4	6	4	1

The formula for getting the k^{th} number (when you start counting at 0) in the n^{th} row is $\frac{n!}{k!(n-k)!}$.

This calculation is known as the combination.

The exclamation point (!) in that formula is used to indicate a factorial; the factorial of any number x is the product of all the integers from 1 up to an including x . As a clarifying example, since $2! = 1 * 2 = 2$ and $4! = 1 * 2 * 3 * 4 = 24$, the 2nd number (when you start counting at 0) of the 4th row is $4! \div (2! (4-2)!) = 24 / (2(2)) = 6$.

You must write a factorial function and a combination function yourself and your combination function must call your factorial function. **You may not use global variables and programs that import the math library or use recursion will not be accepted and will receive a mark of zero.**

Your program will begin by asking the user for an integer value to look for in the triangle – you can safely assume that the user will enter a number but if the user enters a floating-point number your program must print an error message and loop back to allow the user to try and enter another number. On the other hand, if the user gives a valid integer (e.g., 6) then your program will start printing complete rows of the triangle until it encounters a 6 – once it encounters a 6, your program should **complete the row and then terminate**.

Your program must center the triangle (as much as possible, as depicted above) in a console that is 60 characters across and is expected to use the most appropriate type of looping control structure (pre-tested while, post-tested while, or for) for each subproblem.

Question 2 – Chessboard Evaluation with the Relative Value System

For this exercise you will write a program that will assess (numerically) who is winning in a game of chess, using the chess piece relative value system. Sample input and output has been provided below:

```
Please type 8 characters for the 8th row of the chessboard: RNBQKBNR
Please type 8 characters for the 7th row of the chessboard: PPPPPPPP
Please type 8 characters for the 6th row of the chessboard: -----
Please type 8 characters for the 5th row of the chessboard: -----
Please type 8 characters for the 4th row of the chessboard: -----
Please type 8 characters for the 3rd row of the chessboard: -----
Please type 8 characters for the 2nd row of the chessboard: pppppppp
Please type 8 characters for the 1st row of the chessboard: rnbqkbnr
```

White has a score of 40 and Black has a score of 40, so this game is a tie.

```
Please type 8 characters for the 8th row of the chessboard: -----K--
Please type 8 characters for the 7th row of the chessboard: --N-----
Please type 8 characters for the 6th row of the chessboard: -----
Please type 8 characters for the 5th row of the chessboard: -----
Please type 8 characters for the 4th row of the chessboard: ----k---
Please type 8 characters for the 3rd row of the chessboard: --q-----
Please type 8 characters for the 2nd row of the chessboard: -----
Please type 8 characters for the 1st row of the chessboard: -----
```

White has a score of 10 and Black has a score of 3, so White is winning.

You can read more about the relative value system at http://en.wikipedia.org/wiki/Chess_piece_relative_value, but **please note that you must use the values specified in the paragraph below.**

Your program must use lowercase letters for the white pieces and uppercase letters for the black pieces, and your program must use the hyphen "-" for an empty space and the following abbreviations – (K)ing, (Q)ueen, (B)ishop, k(N)ight, (R)ook, and (P)awn. By the chess piece relative value system, kings are not assigned a value, but queens are worth 10 points each, rooks are worth 5 points each, knights and bishops are worth 3 points each, and pawns are worth 1 point each.

Your program will ask the user for each row of the chessboard and receive a string from the user, and by looking through the strings provided for each row, the score for each player can be computed. Your program must create a dictionary that maps the chess piece abbreviations you read from the string into numerical values and use that dictionary to compute the score. **Programs that do not use a dictionary will not be accepted and will receive a mark of zero.**