Computer Organization and Structure

Please write the following three programs in MIPS assembly language.

1. **Combination:**
   Your task is to implement the combination function to compute combination of X and Y, and X, Y should be given by user. For example, given X = 4, Y = 2, then compute $C_Y^X = C_2^4$ and get the answer 6. On the other hand, given X = 2, Y = 4, you should compute $C_X^Y = C_2^4$. If X or Y is a negative integer, please show an error message. Basically you don't have to deal with BIG NUMBER problem, but if you solve it, write the feature in the document and you will get bonus credits.

   Your output should look like this.

   ```
   ----Combination Function---
   Please type 2 integers, and each with the Enter keys.
   4
   2
   6 is the combination results.
   ```

2. **Matrix Transpose:**
   Write a SPIM program that will compute the transpose of the matrix stored in row major order starting at the location labeled `Original` in the starter template below. The transposed matrix should be stored starting at the location labeled `Second`. The arrays we use will be 4x4 elements. Here is the program template to help you get started (The program template is included in the file http://graphics.im.ntu.edu.tw/~robin/courses/arch07/matrix.s).

   ```
   .data
   strA:     .asciiz "Original Matrix:\n"
   strB:     .asciiz "Transpose Matrix:\n"
   newline: .asciiz "\n"
   space :  .asciiz " "

   # This is the start of the original array.
   Original:.word 200, 270, 250, 100
            .word 205, 230, 105, 235
            .word 190, 95, 90, 205
            .word 80, 205, 110, 215

   # The next statement allocates room for the other array.
   # The array takes up 4*16=64 bytes.
   #
   Second:  .space 64
   ```

```
        .globl main

        .text
main:
        # Your fully commented program starts here.
```

Your output should look like this.

```
Original Matrix:
200 270 250 100
205 230 105 235
190 95 90 205
80 205 110 215

Transpose Matrix:
200 205 190 80
270 230 95 205
250 105 90 110
100 235 205 215
```

Tower of Hanoi problem
Your task is to complete a recursive implementation of the Tower of Hanoi problem in order to get familiar with assembly programming. You will need to implement the `mvdsk` function in http://graphics.im.ntu.edu.tw/~robin/courses/arch07/hanoi.s. Refer to the algorithm from http://en.wikipedia.org/wiki/Tower_of_Hanoi.

Your output should look like this.

```
        1
        2
        3
--------------------------------------

        2
        3       1
--------------------------------------


        3       1       2
--------------------------------------

                        1
        3               2
--------------------------------------

                        1
                3       2
--------------------------------------


        1       3       2
--------------------------------------

                2
        1       3
--------------------------------------
                1
                2
                3
--------------------------------------
```

Submission & Grading

Description	For Each Problem
Program compiles without errors	10%
Program executes correctly	60%
Documentation and description of the program	10%
Presentation	10%
Implement Detail	10%