COSC 2436 Lab 1: Recursion

1. **Introduction**

   You will create a C++ program to check if the given integer is possible to be made by a combination of adding adjacent integers in the given matrix. Your program will read inputs files containing a matrix of integers and the target number to be reached. Output "Possible" (without the quotes) if a combination of integers within the matrix can result in the target number. Otherwise, the output is "Not Possible".

2. **Input File**
   - The first input line will contain two integers *m* and *n*, separated by a single space. The integer indicates the number of rows and the number of columns, respectively.
   - The following *m* lines will each contain *n* number of integers; a single space will separate each character.
   - The last line in the input will contain an integer, representing the target number.
   - Each integer in the matrix is unique, meaning that the number will appear only once in the matrix.
   - There can be empty cases, in which the output is "Not Possible"
   - Assumptions:
       - Your starting point is at 0.
       - You can only "add" integers.
       - Each integer in your combination should be unique.
       - No extra lines or unnecessary whitespaces in the input file.

3. **Output File**
   - Output "Possible" if the number can be reached, and "Not Possible" otherwise.

4. **Examples**

   input1.txt
   3 4
   1 2 3 4
   5 6 0 8
   9 10 11 12
   15

   output1.txt
   Possible
   Explanation: 0+3+4+8=15 (our target), 0+6+2+3+4=15 also. There can be multiple combinations. The output is the same as long as there exists one combination.

   input2.txt
   2 3
   0 -3 4
   8 54 -2

52

output2.txt
Not Possible
Explanation: 52 is possible with integers 54 and -2; however, they can't be reached by adjacency to your starting point 0.

input3.txt (empty)

output3.txt
Not Possible

5. **Turn in your lab assignment**
   Lab 1 needs to be turned into our Linux server, follow the link here
   https://rizk.netlify.app/courses/cosc2430/2_resources/
   Make sure to create a folder under your root directory, name it lab1 (the name must be lower case), and only copy your code to this folder, no test cases or other files are needed.
   PS: This document may have typos, if you think something is illogical, please email the TAs for confirmation.