COT 3100 Fall 2024 Project #1: Grid Walking
Please Consult WebCourses for the due date/time

Note: Please *justify* your answers and why you use each formula. Please leave answers in factorials, combinations and powers.

In class, we talked about solving a grid walking problem. For this project, we'll solve one question with several parts by hand and then write a program to solve similar problems. Because the actual answer for these problems can be very high, in your program, you'll report the answers modulo $10^9 + 7$. This will require the use of the type long long in C. If you use Python, no special integer type is needed. In class a similar (but shorter) project example will be done so you can see what is expected of the program, format wise.

1) (15 pts) You live on a Cartesian grid at the point (0, 0) and need to walk to school, which is located at (13, 17) each day. In choosing your walking path, you must always walk either 1 unit in the positive x-axis or 1 unit in the positive y-axis. (Thus, in this case, you walk 30 units, of which you choose any 13 of them to move in the direction of the positive x-axis.) Unfortunately, there may be some bullies on the way that you may want to avoid, so in the following questions, we'll find the number of paths from home to school under a varying set of restrictions. **Please leave your answers in terms of combinations, factorials, etc.**

a) No restriction other than what is listed above.

b) You need to pick up your friend on the way to school who is located at (6, 4).

c) A second friend has asked to be picked up who lives at (9, 14), so you must first pick up your friend at (6, 4) and then proceed to (9, 14) before going to school.

d) A third friend asks to be picked up who lives at (10, 1), in addition to those who are at (6, 4) and (9, 14).How many ways can all three friends be picked up still adhering to our walking rules? (This one is very easy.)

e) Now, there are no friends to pick up, but bullies have infiltrated the neighborhood.
There is a bully at (3, 9), so this location must be avoided.

f) The bully at (3, 9) has a friend who has placed himself at (10, 13), so now you must avoid both locations.

2) (85 pts) Write a program that will answer two types of grid walking queries that are described in the examples above.

(a) Given a starting location, an ending location, upto 5 positions to visit, determines the number of ways to go from start to end visiting each of the locations, modulo 10^9+7. Note: you must handle the special case indicated in problem 1d.)

(b) Given a starting location, an ending location, and one or two positions to avoid, determines the number of ways to go from start to end avoiding the given intermediate positions, modulo 10^9+7. (This also has a special case similar to 1d that must be properly coded.)

Your program will run without prompting the user but will read input from standard input and produce output to standard output as specified below. **Please write your program in C or Python.** Your program will be tested on multiple sets of input.

An example will be shown in class to clarify the format.

Input Format

The first line of input will contain two positive integers, x ($5 \leq x \leq 100$) and y ($5 \leq y \leq 100$), indicating where school is located. (Home is still at (0, 0).)

The second line of input will contain a single positive integer, q ($1 \leq q \leq 100$), indicating the number of queries to solve for the given starting and ending location on the first line.

Each scenario will consist of multiple lines. The first line of each scenario will have a single positive integer, t ($t = 1$ or $t = 2$), indicating the type of scenario. If $t = 1$, it's the type of scenario where we must visit several locations on our way to school. If $t = 2$, it's the type of scenario where we must avoid one or two locations on our way to school.

For both types of scenarios, the second line of each scenario will have a single positive integer, n, either indicating the number of locations that must be visited (if $t = 1$), or the number of locations that must be avoided (if $t = 2$). Note: if $t = 1$, then $1 \leq n \leq 5$, and if $t = 2$, then $1 \leq n \leq 2$.

The following n lines will each have a pair of ordered integers x_i and y_i, indicating that location i ($1 \leq i \leq n$) is at (x_i, y_i). To make organizing the information easier, it will be guaranteed that $0 < x_1 < x_2 < ... < x_n < x$. The same requirements aren't true for the y values, but all of the y_i's will be greater than 0 and less than y.

Sample Input

```
13 17
5
1
1
6 4
1
2
6 4
9 14
1
3
6 4
9 14
10 1
2
1
3 9
2
2
3 9
10 13
```

Sample Output

```
16279200
2102100
0
110133090
72631780
```