Project 3: Graph modeling and graph algorithms

COT 4400, Spring 2025

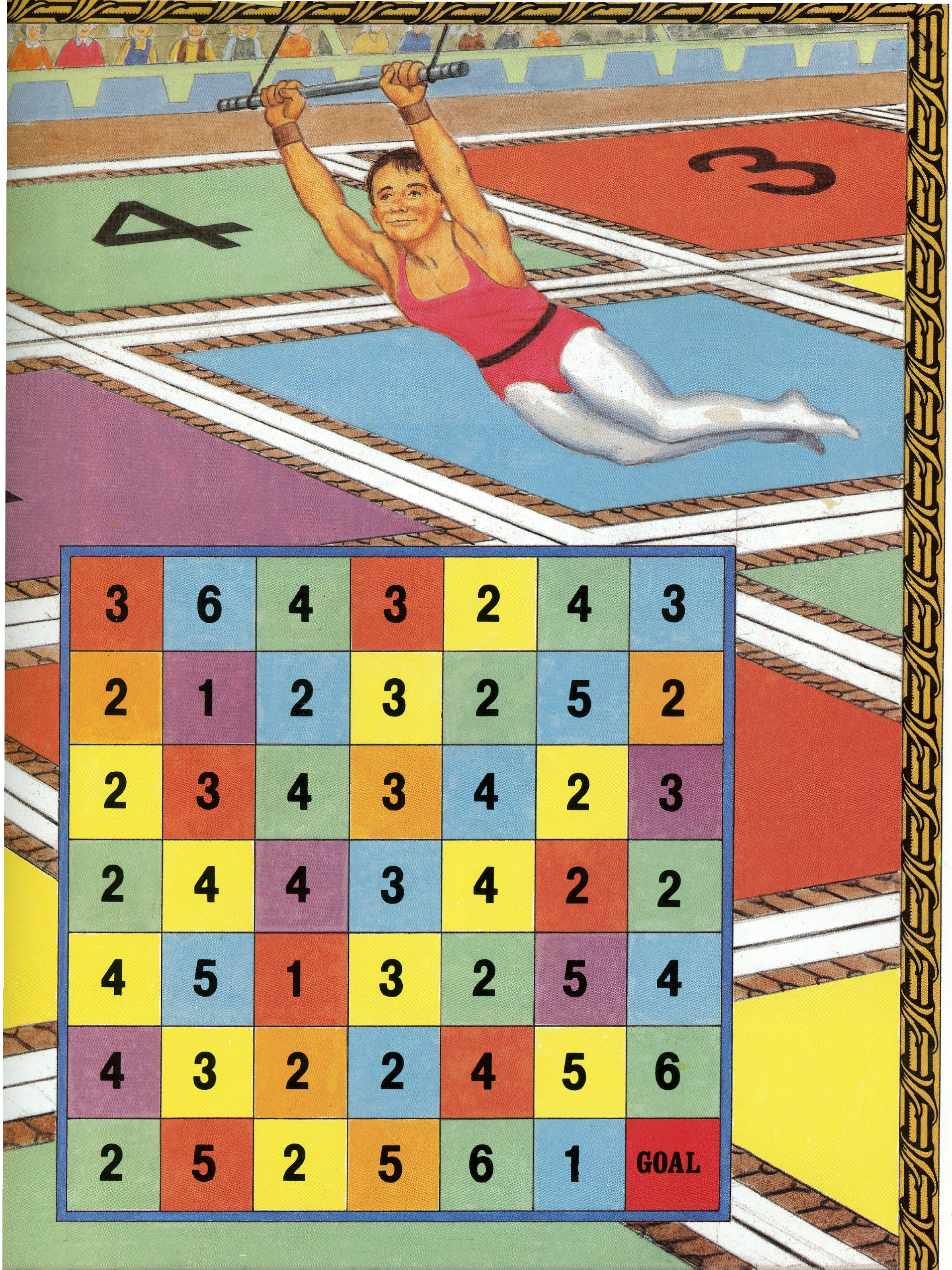
Due May 7, 2025

1 Overview

This project requires you to model the problem below as graph and then use a known graph algorithm to solve the problem. **You are not allowed to use the internet or consult any references. This is an individual project. This policy will be strictly enforced.**

This problem is based on the "Jumping Jim" maze problem (from "MAD MAZES: Intriguing Mind Twisters for Puzzle Buffs, Game Nuts and Other Smart People" by Robert Abbott). The text of the problem is quoted below. A diagram of the maze is provided on the next page.

> Jumping Jim is about to begin his grand performance at the circus, but his jealous enemy, Dastardly Dan, has restrung all the trampolines. The number on each trampoline indicates how tightly strung each one is; in other words, the number indicates how far Jim will have to move (horizontally or vertically, but NOT diagonally) when he bounces off the trampoline. Jim begins his routine by leaping onto the trampoline at the upper left. He must get to the Goal at the lower right, where he will take his bow. How can he get there?
>
> The diagram shows all of the trampolines with each represented as a square. Begin on the square at the upper left. That square is marked 3. From there you could, for example, move three squares down to a square marked 2. From there, you might move two squares right to a square marked 4, and from there you could move four squares right to another square marked 2. That path, incidentally, won't get you to the goal.



3	6	4	3	2	4	3
2	1	2	3	2	5	2
2	3	4	3	4	2	3
2	4	4	3	4	2	2
4	5	1	3	2	5	4
4	3	2	2	4	5	6
2	5	2	5	6	1	GOAL

2 Modeling the problem

Before you write a program to solve this problem, you will first write a report describing (in English and pseudocode) how you will solve this problem. This report should answer two basic questions:

1. What type of graph would you use to model the problem input (detailed in the Section 3.1), and how would you construct this graph? (I.e., what do the vertices, edges, etc., correspond to?) Be specific here; we discussed a number of different types of graphs in class.

2. What algorithm will you use to solve the problem? Be sure to describe not just the general algorithm you will use, but how you will identify the sequence of moves Jim must take in order to reach the goal.

3 Coding your solution

In addition to the report, you should implement your algorithm in C++ or Java so that it can solve "Jumping Jim" mazes. Your code may be in C++ or Java.

Your code may be split into any number of files. In addition, you are allowed to make use of any built-in library, and C++ users may use the Boost library in their implementations. Boost is a free, open-source library with a rich collection of mathematical functions, including several that deal with graphs. You may read more about Boost at `www.boost.org`.

3.1 Input format

Your program should read its input from the file `input.txt`, in the following format. The input begins with a two positive integers on a line indicating the number of rows r and columns c of the maze, respectively. The next r lines detail all of the trampolines in the maze. Each line has c integers, where each integer represents the number of spaces Jim will move when landing on that trampoline. The goal square will always be in the lower right of the maze.

For the original "Jumping Jim" maze, the input is:

```
7  7
3  6  4  3  2  4  3
2  1  2  3  2  5  2
2  3  4  3  4  2  3
2  4  4  3  4  2  2
4  5  1  3  2  5  4
4  3  2  2  4  5  6
2  5  2  5  6  1  0
```

3.2 Output format

Your program should write its output to the file `output.txt`, in the following format. The output should consist of a path from the top left square to the bottom right square. It should be a single line consisting of a sequence of moves, separated by spaces, where each move is a single letter representing the direction of that move: N, E, S, or W. Of course, the sequence of moves described in the file must solve the maze described in the input.

For example, if your first three moves take you 3 spaces right, 3 spaces down, and 2 spaces right, your output should begin with `E S E`.

You are welcome to try figuring out the solution to the "Jumping Jim" puzzle on your own, but that won't get you any points. Your assignment is to model the maze as a graph and to solve the problem using an appropriate graph algorithm.

4 Submission

You must submit a zip archive containing 1) your report (described in Section 2) as a PDF document, 2) your code (described in Section 3), and 3) a README file describing how to compile and run your code to Canvas. If your code is in Java, the `main` function should be located in `JumpingJim.java`. If your code requires more than a simple command to compile and run then you must also provide a Makefile and/or shell script. A simple command might be something like:

```
g++ *.cpp -o jumpingjim
```

or:

```
javac *.java
```

```
java JumpingJim
```

If you are using Boost in your solution, you must provide a Makefile and/or shell script that uses the environment variable `$BOOST_HOME` (pointing to the Boost installation directory) to compile your code.

As this is an *individual* project, your project report and code will be checked for plagiarism.

5 Grading

Report	**50 points**
Graph model	30
Algorithm description	20
Code	**50 points**
README file	5
Follows input and output specs	10
Compiles and is correct	30
Good coding style	5

Note: if your code is unreasonably slow, you will lose points for both your algorithm design and your correct output grade.