Project 2: Dynamic programming

COT 4400, Fall 2019

Due November 20, 2019

1 Overview

For this project, you will develop an algorithm to find a maximum sum while travelling through a matrix. Designing and implementing this solution will require you to model the problem using dynamic programming, then understand and implement your model.

You are only allowed to consult the class slides, the textbook, the TAs, and the professor. In particular, you are not allowed to use the Internet. This is a group project. The only people you can work with on this project are your group members. This policy is strictly enforced.

In addition to the group submission, you will also evaluate your teammates' cooperation and contribution. These evaluations will form a major part of your grade on this project, so be sure that you respond to messages promptly, communicate effectively, and contribute substantially to your group's solution. Details for your team evaluations are in Section 6.2. You will submit the peer evaluations to another assignment on Canvas, labelled "Project 2 (individual)."

A word of warning: this project is team-based, but it is quite extensive and a nontrivial task. You are highly encouraged to start working on (and start asking questions about) this project early; teams who wait to start until the week before the due date may find themselves unable to complete it in time.

2 Problem Description

In this problem, you are given a matrix of positive integers, and your goal is to maximize a sum by selecting one element from every column in the matrix, moving left-to-right. As you move through the matrix column-by-column, though, there may be a penalty to your sum depending on how you move relative to your previous two positions. If the next row you select is between the previous two selected rows, there is no penalty; however, there is a penalty of 2 to your sum for every row above the maximum of the previous two or below the minimum of the previous two. You always start in the first column of the matrix and your previous two rows are considered to be the top row.

For example, if you start with row 3 in column 1 of the matrix, there will be a penalty of 4, since 3 is 2 more than $\max(1, 1)$. If you then move to row 2 in column 2, there would be no penalty because 2 is between 1 and 3. If you then moved to row 7, there would be a penalty of 8 because 7 is 4 more than the max of 2 and 3.

The total "score" of the elements you select from the matrix will be the sum of the elements you selected, minus any penalties you accrued, and your algorithm should find the maximum possible score for the given matrix.

3 Example problem

Consider the 3×4 matrix $\begin{bmatrix} 2 & 3 & 4 & 1 \\ \mathbf{5} & 1 & 2 & 4 \\ 4 & \mathbf{5} & \mathbf{3} & \mathbf{4} \end{bmatrix}$. The best score we can make for this matrix is 13 by choosing elements 5, 5, 3, and 4 (highlighted):

Column	Last 2 rows	"Free" rows	Best element (row)	Penalty	Score
1	1 and 1	1	5 (2)	2	3
2	2 and 1	1, 2	5 (3)	2	6
3	3 and 2	2, 3	3 (3)	0	9
4	3 and 3	3	4 (3)	0	13

It's also possible to make a score of 12 with this matrix by choosing elements 5, 3, 4, and 4 (rows 2, 1, 1, and 2) or by choosing 5, 5, 4, 4 (rows 2, 3, 1, 2), but neither of these routes is optimal:

Column	Last 2 rows	"Free" rows	Element (row)	Penalty	Score
1	1 and 1	1	5 (2)	2	3
2	2 and 1	1, 2	3 (1)	0	6
3	1 and 2	1, 2	4 (1)	0	10
4	1 and 1	1	4 (2)	2	12

Column	Last 2 rows	"Free" rows	Element (row)	Penalty	Score
1	1 and 1	1	5 (2)	2	3
2	2 and 1	1, 2	5 (3)	2	6
3	3 and 2	2, 3	4 (1)	2	8
4	1 and 3	1, 2, 3	4 (2)	0	12

4 Project report

In your project report, you should include brief answers to 8 questions. Note that **you must use dynamic programming** to solve this problem; other solutions, such as those based on graph traversals, will not receive substantial credit.

1. How you can break down a problem instance of finding a maximum score for a matrix A with n rows and m columns, where your previous two selected rows are r_{old} and r_{new} into smaller instances of the same problem? Your answer should include how you calculate the maximum score for the original problem based on the solution(s) to the subproblem(s).

2. What are the base cases of this recurrence?

3. What data structure would you use to recognize repeated problems? You should describe both the abstract data structure, as well as its implementation.

4. Give pseudocode for a memoized dynamic programming algorithm to find the maximum score for a matrix A with n rows and m columns, with previously selected rows of r_{old} and r_{new}.

5. What is the *worst-case* time complexity of your memoized algorithm?

6. Give pseudocode for an iterative algorithm to find the maximum score for a matrix A with n rows and m columns, with previously selected rows of r_{old} and r_{new}. This algorithm does not need to have a reduced space complexity relative to the memoized solution.

7. Describe at least one advantage and disadvantage of the iterative algorithm, and indicate which implementation you believe is better overall.

8. Give pseudocode for an algorithm that identifies which entries (rows) to select in each column in order to achieve the maximum score. Your algorithm may be iterative or recursive.

5 Coding your solutions

In addition to the report, you should implement a dynamic programming algorithm that can find the maximum score for a given matrix. Your code may be iterative or recursive, but it must be a dynamic programming algorithm. Also, you may code your solution in C++ or Java, but it must compile and run in a Linux environment. If you are using C++ and compiling your code cannot be accomplished by the command

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

you should include a Makefile that is capable of compiling the code via the `make` command.

If you choose to implement your code in Java, you should submit an executable jar file with your source. In either case, your source code may be split into any number of files.

Your code will not need to handle invalid input (e.g., a matrix with a negative number of rows or columns).

5.1 Input format

Your program should read its input from the file `input.txt`, in the following format. The first line of the file has two positive integers n and m specifying the number of rows and columns of the matrix, respectively. The following n lines will have m positive integers, representing the m entries in that row of the matrix.

5.2 Output

Your program should write its output to the file `output.txt`. Your code should first print out the maximum score on a line. Then, your program should print out m integers on the following line, representing which row in each column that you should select to reach that maximum score. Each row number should be in the range 0 to $n - 1$, and the rows should be specified in left-to-right order.

6 Submission

Your submission for this project will be in two parts, the group submission and your individual peer evaluations.

6.1 Group submission

The submission for your group should be a zip archive containing 1) your report (described in Section 4) as a PDF document, and 2) your code (described in Section 5). If your code requires

more than a simple command to compile and run then you must also provide a Makefile and/or shell script. You should submit this zip archive to the "Project 2 (group)" assignment on Canvas.

Be aware that your project report and code will be checked for plagiarism.

6.2 Teamwork evaluation

The second part of your project grade will be determined by a peer evaluation. Your peer evaluation should be a text file that includes 1) the names of all of your teammates (including yourself), 2) the team member responsibilities, 3) whether or not your teammates were cooperative, 4) a numeric rating indicating the proportional amount of effort each of you put into the project, and 5) other issues we should be aware of when evaluating your and your teammates' relative contribution. The numeric ratings must be integers that sum to 30.

It's important that you be honest in your evaluation of your peers. In addition to letting your team members whether they do (or do not) need to work on their teamwork and communication skills, we will also evaluate your group submission in light of your team evaluations. For example, a team in which one member refused to contribute would be assessed differently than a team with three functioning members.

You should submit your peer evaluation to the "Project 2 (peer eval)" assignment on Canvas.

7 Grading

Report	**40 points**
Question 1, 4, and 6	8 points
Questions 2 and 3	2 each
Questions 5, 7, and 8	4 each
Code	**30 points**
Compiles	5
Uses correct input and output format	5
Computes correct answer	15
Good coding style	5
Teamwork	**30 points**

Note that if your algorithm is inefficient, you may lose points for both your pseudocode and your submission. Also, in extreme cases, the teamwork portion of your grade may become negative or greater than 30. In particular, if you do not contribute to your group's solution at all, you can expect to receive an overall grade of 0 on the project.