Precept Outline

- Review of Lectures 19 and 20:
  - Dynamic Programming
  - Maxflow/Mincut and Ford-Fulkerson

Relevant Book Sections

- Book chapters: 6

A. Review: Dynamic Programming and Maxflow/Mincut

Your preceptor will briefly review key points of this week's lectures.

B. Dynamic Programming

This question was adapted from the Fall'23 Final Exam.

Suppose you are taking an idealized exam with n questions and have m minutes to complete it. Question j is worth p_j points and takes t_j minutes to earn the points. Your goal is to maximize the number of points earned in the allotted time. Assume that all p_j and t_j are positive integers (and that there is no partial credit in this idealized exam).

You will solve this problem using dynamic programming. Let's start by defining some meaningful overlapping subproblems. For each i and j with $0 \leq i \leq m$ and $0 \leq j \leq n$, give a definition of OPT(i, j) that you think might be useful.

OPT$(i, j) =$

Given this definition, fill in the following blanks:

OPT$(0, j) =$

OPT$(i, 0) =$

OPT$(m, 1) =$

Recall that your goal is to maximize the number of points earned in the allotted time. Give the solution to the problem in terms of some value of OPT(i, j)

Now we need to compute the value of OPT(i, j) in order to solve the problem. Define a recurrence relation to find OPT(i, j) given smaller values of i, j.

$$\text{OPT}(i,j) = \begin{cases} & \text{if} \\ & \text{if} \end{cases}$$

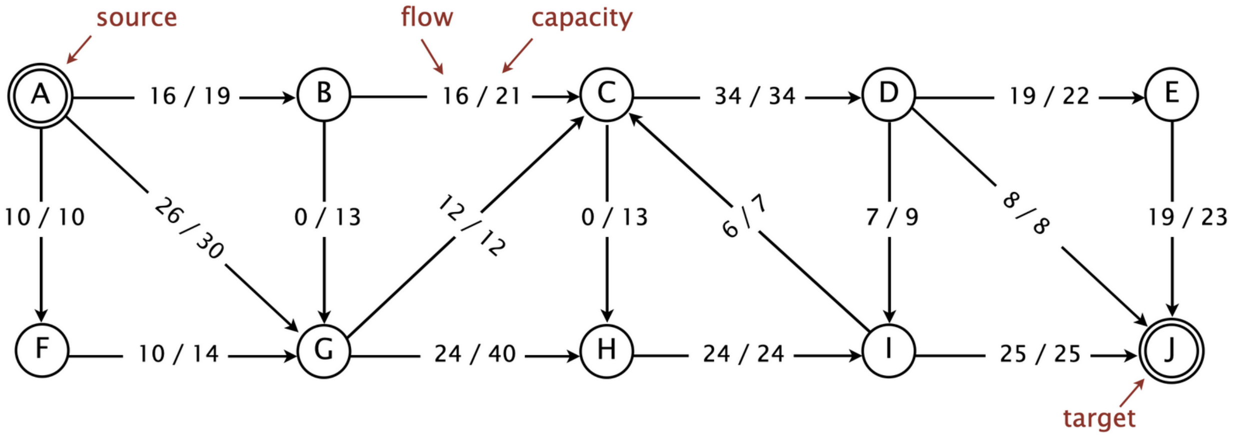
Now, put everything together. Fill in the following Java code to solve this problem.

```java
int maximumExamScore(int m, int n, int[] points, int[] times) {
    int[][] opt = new int[m+1][n+1];
    for (int i = 1; i <= m; i++) {
        for (int j = 1; j <= n; j++) {
            if (times[j] > i)
                opt[i][j] = _____________________________________________________;
            else
                opt[i][j] = _____________________________________________________;
        }
    }
    return opt[m][n];
}
```

Finally, what is the running time of this algorithm?

C. Maximum Flow

Consider the following flow network with a flow f from source vertex A to sink vertex J.



What is the value of f? Compute the *capacity* of the cut $\{A, B, C\}$ and the net *flow* across this cut.

Find the maximum flow in the network.

Find a minimum cut in the network. Which vertices are on the source side of the cut?