Lab 12 - Dynamic Programming

Solve the following problems using dyamic programming:

1. Compute the n-th number of the Fibonacci sequence. ★

2. Given a rod of length `n` and an array `price`, where `price[i]` denotes the value of a piece of length i, determine the maximum value obtainable by cutting up the rod and selling the pieces. ★★

3. Given an n×n matrix of positive or negative integers find the path from the top left corner (0, 0) to the bottom right corner (n-1, n-1) that has the maximum sum. You can only move down or right. ★★

4. A man wants to climb a stair that consists of n steps. At each step he can go up 1, 2 or 3 steps. Every m-th step of the stair is missing (the man cannot step on it). Find out in how many ways the stair can be climbed for any n or m. ★★

5. Given a matrix of $n \times n$ numbers find the number of paths from any position on the first row to any position on the last row where the numbers along the path are strictly increasing. You can only move one position down, one down and one right or one down and one left. ★★

6. Given n items, each with value v_i and weight w_i, and a knapsack with capacity W, select items to maximize the total value without exceeding the capacity. ★★★

Note: Leave a comment with the text PB1, PB2.A.II, ... PB10 above every function that implements the respective lab task. (upper case text, no space between the text and the problem number)