CS361 Final
Due: Friday, December 13th at 11:59pm

Cutting a Rod with Dynamic Programming

The Problem:

Given a rod of length **n,** and an array of prices **P**, determine the maximum possible revenue from cutting the rod and selling its pieces.

Edge Cases:

It is possible for the highest sale price to come from no cuts.

It is also possible that you would be given a rod of length zero, in which case your maximum possible revenue should also be 0.

Finally, it is possible that you would be given a nonzero rod and an array of prices that contains lengths longer than your rod, in which case your function should be able to work on only those prices for rod lengths you can actually have.

Your Task:

You will write a C++ program that implements a function cutRod( ) with ***dynamic programming***, that is, a function that stores solutions to its subproblems in a table.

Your cutRod( ) function should take in at least an array of prices **P** = [p1, p2, …] and a rod length **n,** where all values are integers. The length for a given price is its position in the array--meaning p1 corresponds to a segment of length 1, p2 a segment of length 2, etc.

Depending on whether you implement a top-down (memoized) or a bottom-up (tabulated) solution, you function may take additional input (e.g., an array of solutions r) as needed, or call a helper function that “hides” the solution array from your main cutRod( ) function.

In either case, your cutRod( ) function should only return an integer value, the maximum revenue from cutting and selling the rod.

For grading, please print at least two test case inputs and outputs to the console.

Example Console Output:

```
>>cutting rod of length 4, selling prices are [1, 5, 8, 9, 10, 17, 17, 20]
>>maximum revenue is 10
>>
>>cutting rod of length 0, selling prices are [1, 5, 8, 9, 10, 17, 17, 20]
>>maximum revenue is 0
>>
>>cutting rod of length 4, selling prices are [0, 0, 8, 9, 10, 17, 17, 20]
>>maximum revenue is 8
>>
>>
```

Helpful Resources:

It is highly advised that you attempt to solve this problem on your own, and pair programming is not permitted on this assignment. If you get stuck and frustrated, I suggest you walk away from the computer, drink a glass of water, and do something else for 10 minutes before returning and attempting to draw your program's execution.

If all else fails, the following is a helpful guide that explains as much as possible about the problem in steps, without spoiling future steps or providing an explicit code solution:
https://www.radford.edu/~nokie/classes/360/dp-rod-cutting.html

May the odds be ever in your favor.