Dynamic Programming

1. You are a professional robber planning to rob houses along a street. Each house has a certain amount of money stashed, which you are able to make an accurate estimate of based on publicly available information on the internet. You need to decide which houses to rob, where the only constraint stopping you from robbing each of them is that adjacent houses have security systems connected and it will automatically contact the police if any two adjacent houses were broken into on the same night. (You may assume all houses have positive values.)

   (a) Give a (small) example of why choosing the largest value house first (in other words, being greedy) will not necessarily yield the most money.

   (b) Write a recursive formula or expression for the maximum amount of money you can gain.

   (c) Given an array Houses$[1 \ldots n]$, where Houses$[i]$ is the amount of money estimated in the i^{th} house, design an algorithm to calculate the maximum amount of money you can rob tonight without alerting the police.