Homework

due April 17, 2020 via dropbox as a pdf

Problem 1:

Develop a dynamic programming solution for the following problem: You are given a set of positive numbers. Is it possible to divide the set into two disjoint subsets such that the sum of the of numbers in each subset is equal. For example, if the set is $\{1,2,3,4,5,6,7,10\}$ then a successful partition is $\{1,3,5,7\} \cup \{2,4,10\}$.

Hint: Clearly, the sum S of all of the elements in the set needs to be even. If we can find a subset that has a sum of $S/2$, then the complement set also has sum $S/2$. Thus, the problem reduces to finding a subset of a set whose sum is equal to a certain number.

Deliverables: Description of the dynamic programming approach, Python code

Problem 2: A very frustrating exam

Assume you have to take an exam in a subject that can be very frustrating. You have to solve the problems one by one in order, but you can skip problems. You are capable of solving any problem on the exam, but some problems are very frustrating. Problem i has a number of points p_i and a number f_i of problems that need to be skipped if you solve it. Find an algorithm that determines the best number of points possible

EXAMPLE:

Problem	Points	Frustration
1	5	0
2	10	1
3	7	2
4	12	1
5	15	2
6	17	1
7	20	3
8	5	0
9	10	1
10	13	1

Assume that you are given the information in the following table. If you pick Problem 1 and Problem 2, you get 15 points, but now you have to skip Problem 3. If you pick Problem 4, then you get 12 points, but you miss out on Problem 5. If you skip Problem 6 and go for Problem 7, you get 20 points and you are at the end of the exam. This strategy will get you 48 points.

Now, clearly, you cannot do wrong with doing Problem 1. But should you include Problem 2 or not?