# AD – Assignment 2

Dynamic Programming

Søren Dahlgaard

Hand-in date: May 22nd

Should be done in groups of 2–3 students

## 1 Motivation

The goal of this assignment is to understand the aspects of dynamic programming: Optimal substructure and overlapping subproblems. Dynamic programming is a very important tool, and being able to understand the recursive nature of such problems can greatly reduce running times.

## 2 Problem statement

Three students at University of Copenhagen have just graduated and want to celebrate. To do this they wish to drink a lot of beer, but first they have to buy it. The local store features $n$ different beers from the CPS Brewing Company. The $i$th beer has a price of $p_i$ DKK and the students have $C$ total DKK to spend. They now wish to know in how many ways they can spend exactly $C$ DKK buying beers and have asked you to help calculating this.

## 3 Example

Consider the following example with $n = 5$ beers and $C = 5$ DKK. The different beer prices are

$$p_1 = 2 \quad p_2 = 3 \quad p_3 = 2 \quad p_4 = 1 \quad p_5 = 4$$

Out of the $2^n$ possible subsets of beer only the following have a total price of 5:

$$(p_1, p_2), \quad (p_1, p_3, p_4), \quad (p_2, p_3), \quad (p_4, p_5),$$

so the total number of ways the students can spend all their money is 4.

## 4 Assignment

**Task 1:** Write a recursive formula for the number of ways the students can spend all their $C$ DKK.

*Hint: You can look at a formula, where $N(C, i)$ denotes the number of ways to spend exactly $C$ DKK on beers with prices $p_1, \ldots, p_i$.*

**Task 2:** Prove that your recursive formula is correct and that it consists of overlapping subproblems.

**Task 3:** Turn your recursive formula into an $O(nC)$ dynamic programming algorithm. Provide pseudocode for the algorithm. You can use either memorization or bottom-up DP.

**Task 4:** Prove the correctness and running time of your algorithm.

What is the memory usage of your algorithm?