Student Number   **Name**

1.   Consider the following problem of "making change", given an unlimited supply of coins with various values:

Input: A positive integer *amount* A and positive integer *denominations* $d_1 < d_2 < \cdots < d_m$. (For example, using pennies, dimes, and quarters, we have $d_1 = 1$, $d_2 = 10$, and $d_3 = 25$.)

Output: A list of coins $c_0, c_1, \ldots, c_{n-1}$ where each $c_i \in \{d_1, d_2, \ldots, d_m\}$ and *repeated coins are allowed*, such that $c_0 + c_1 + \cdots + c_{n-1} = A$ and n is as small as possible. (For example, making change for amount 30 is done with $n = 3$ and $c_0 = 10$, $c_1 = 10$, $c_2 = 10$.)

Define an array that could be used to solve the general problem of making change using a dynamic programming algorithm. (Your solution should work for **any** denominations.) Then, give a recurrence relation for the values in your array, including a brief justification that your recurrence is correct. *Do not write any algorithm* — we want just the recurrence along with a brief English explanation.