Project 1 (20 pt.)

Submission instructions:

This is an **individual** project, and each student must implement the codes independently. (Project 2 will be teamwork of at most 2 students.)

In your first submission attempt in Blackboard, upload a zip file including your source codes and executable file.

In your second submission attempt, upload a pdf file as project report.

The project report should have the following parts: (1) pseudo codes of your dynamic programming algorithm. (2) Analysis of the running time asymptotically. (3) Grouping results of several input examples including the one that A={3,9,7,8,2,6,5,10,1,7,6,4} and M=3. (4) Source codes.

Problem Description:

You are given an input array $A[1, \dots, N]$. A grouping of the array A is described by an array $G[1, \dots, M]$, where the array A is partitioned into M groups, the 1st group consists of the first $G[1]$ elements of array A, the 2nd group consists of the next $G[2]$ elements, and so forth. Define array $B[1, \dots, M]$ such that $B[j]$ is the summation of the elements in the j-th group of array A. Use a dynamic programming algorithm to find a grouping of array A with M groups such that we maximize the minimum element of array B.

Max-min-grouping(A, N, M)
{

return $G[1, \dots, M]$
}

Hint:

- **The optimal subproblem property**: suppose the optimal solution to Max-min-grouping(A, N, M) is $G[1, \dots, M] = [n_1, n_2, \dots, n_{M-1}, n_M]$. Then $G[1, \dots, M-1]$ is the optimal solution to the subproblem Max-min-grouping$(A, N - n_M, M - 1)$.
- See Algorithm 2 in the paper.