

1 Problem Definition

Problem1: Given an array A of n integers (positive or negative), find a contiguous subarray whose sum is maximum.

Problem2: Given a two-dimensional array M of size $m \times n$ consisting of integers (positive or negative), find a rectangle (two-dimensional sub-array) whose sum is maximum.

2 Algorithm Design Tasks

Alg1 Design a $\Theta(n^3)$ time brute force algorithm for solving Problem1.

Alg2 Design a $\Theta(n^2)$ time dynamic programming algorithm for solving Problem1.

Alg3 Design a $\Theta(n)$ time dynamic programming algorithm for solving Problem1.

Alg4 Design a $\Theta(n^6)$ time brute force algorithm for solving Problem2.

Alg5 Design a $\Theta(n^4)$ time algorithm for solving Problem2 using dynamic programming Alg3.

Alg6 Design a $\Theta(n^3)$ time algorithm for solving Problem2 using dynamic programming Alg3.

3 Programming Tasks

Implement each of the following programming procedures:

Task1 Give an implementation of Alg1.

Task2 Give an implementation of Alg2.

Task3a Give a recursive implementation of Alg3 using Memoization.

Task3b Give an iterative BottomUp implementation of Alg3.

Task4 Give an implementation of Alg4 using $O(1)$ extra space.

Task5 Give an implementation of Alg5 using $O(mn)$ extra space.

Task6 Give an implementation of Alg6 using $O(mn)$ extra space.