Algorithms 2018

Homework #2

- Consider the following 4 x 4 array.

	0	1	2	3
0	3	–3	–2	–1
1	2	6	–5	1
2	–3	–5	2	4
3	2	4	0	–2

Homework #2

- The sum of numbers in the following gray rectangle becomes 2 (=6-5-5+2+4+0).

3	–3	–2	–1
2	6	–5	1
–3	–5	2	4
2	4	0	–2

Homework #2

- The sum of numbers in the following gray rectangle becomes 5 (=1+4).

3	–3	–2	–1
2	6	–5	1
–3	–5	2	4
2	4	0	–2

Homework #2

- The sum of numbers in the following gray rectangle becomes 8 (=2+6).

3	–3	–2	–1
2	6	–5	1
–3	–5	2	4
2	4	0	–2

Homework #2

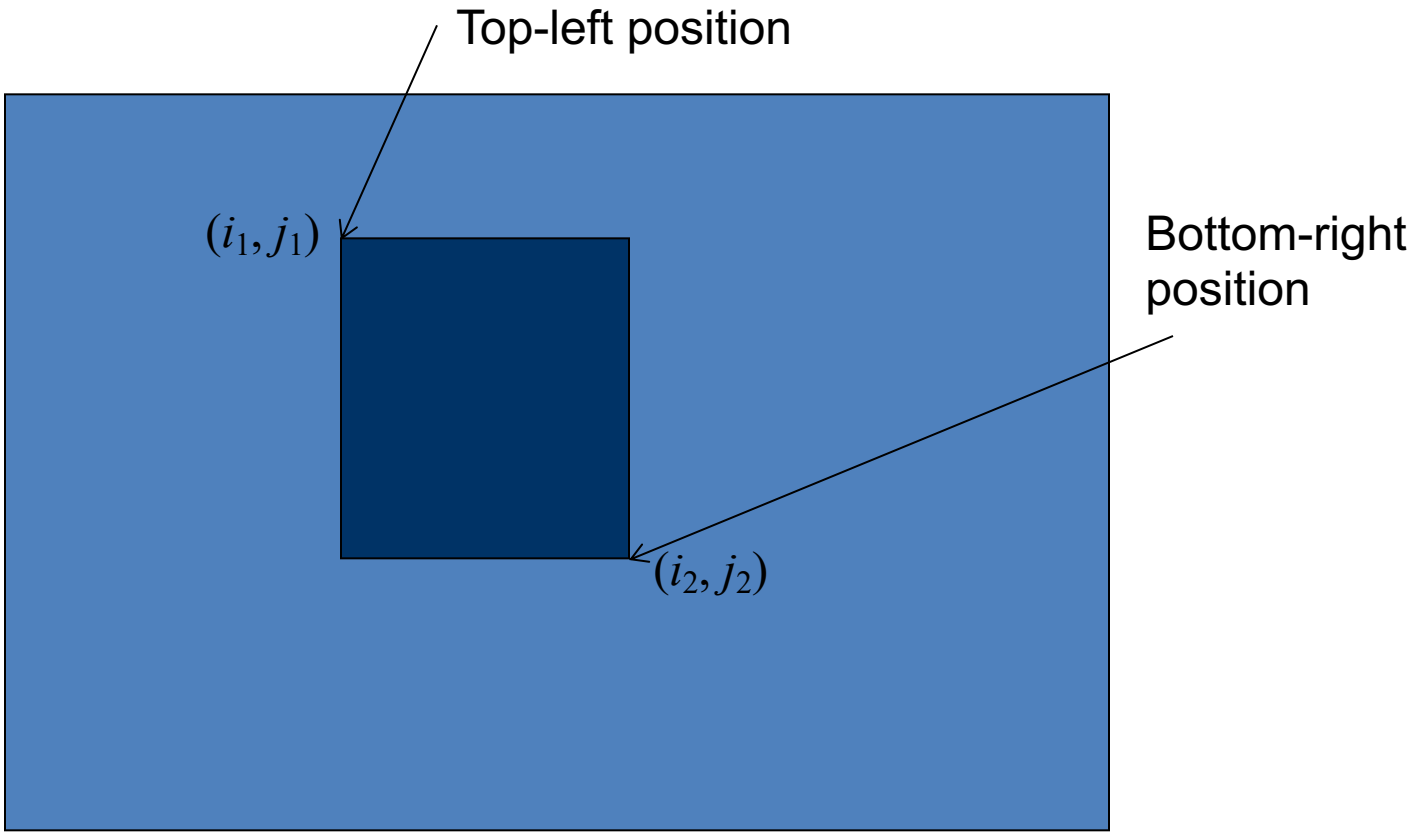
- The sum of numbers in the following gray rectangle becomes -8 (=-3-5).

3	–3	–2	–1
2	6	–5	1
–3	–5	2	4
2	4	0	–2

Homework #2

- Given an n x n integer array, your program should find a rectangle such that the sum of numbers in the rectangle is maximized/minimized.
- Your program should output the maximum/minimum summation value in the first line, and in the next line, the top-left and bottom-right positions of the rectangle that gives the maximum/minimum summation value.
- Note that n is less than or equal to **1,000**.

Homework #2



Examples (1/4)

- Input (standard/console input)

```
5
 15  -3 -66  78  39
  3 -55   1 -91   6
-65  37 -84  52 -99
 37 -50  -8   9 -22
 13  66 -23  58  10
```

n

n x n input array

- Output (standard/console output)

```
124
(4,0) (4,4)
…
(…) (…)
```

Maximum summation value

A rectangle

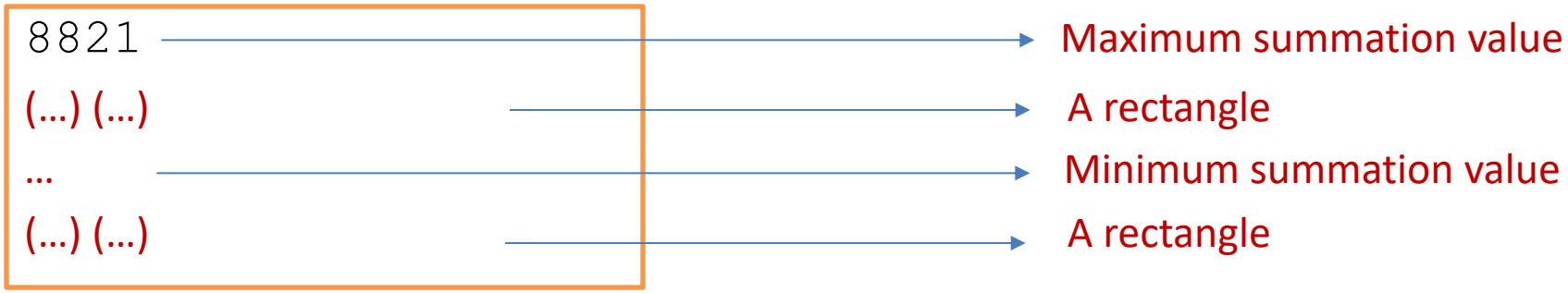
Minimum summation value

A rectangle

Examples (2/4)

- Input (standard/console input)
  - testcase100.txt

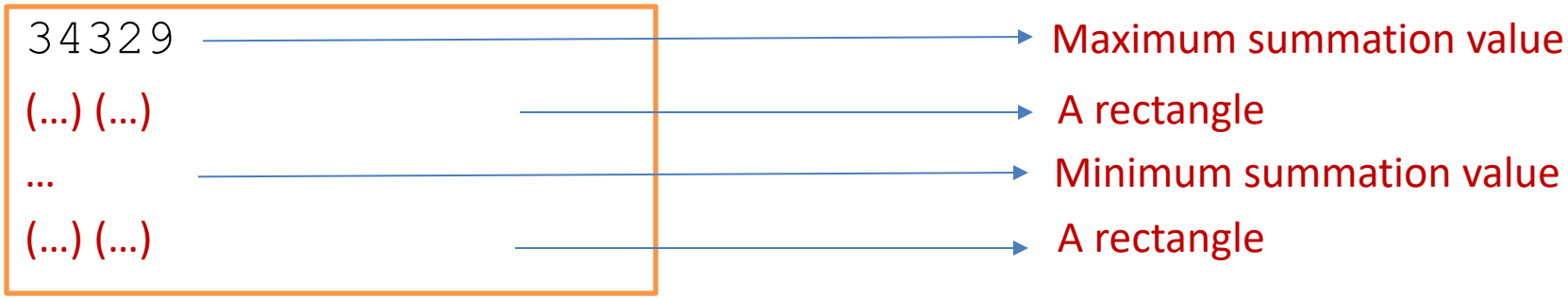
- Output (standard/console output)



Examples (3/4)

- Input (standard/console input)
  - testcase500.txt

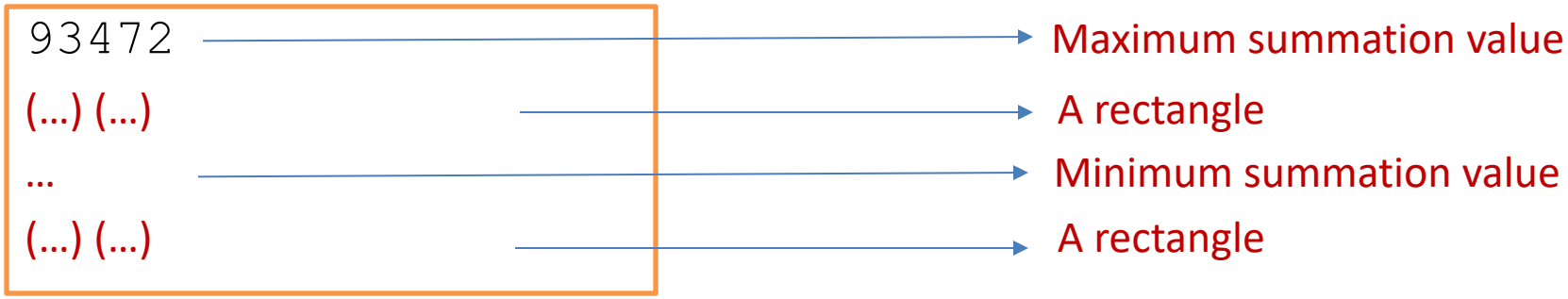
- Output (standard/console output)



Examples (4/4)

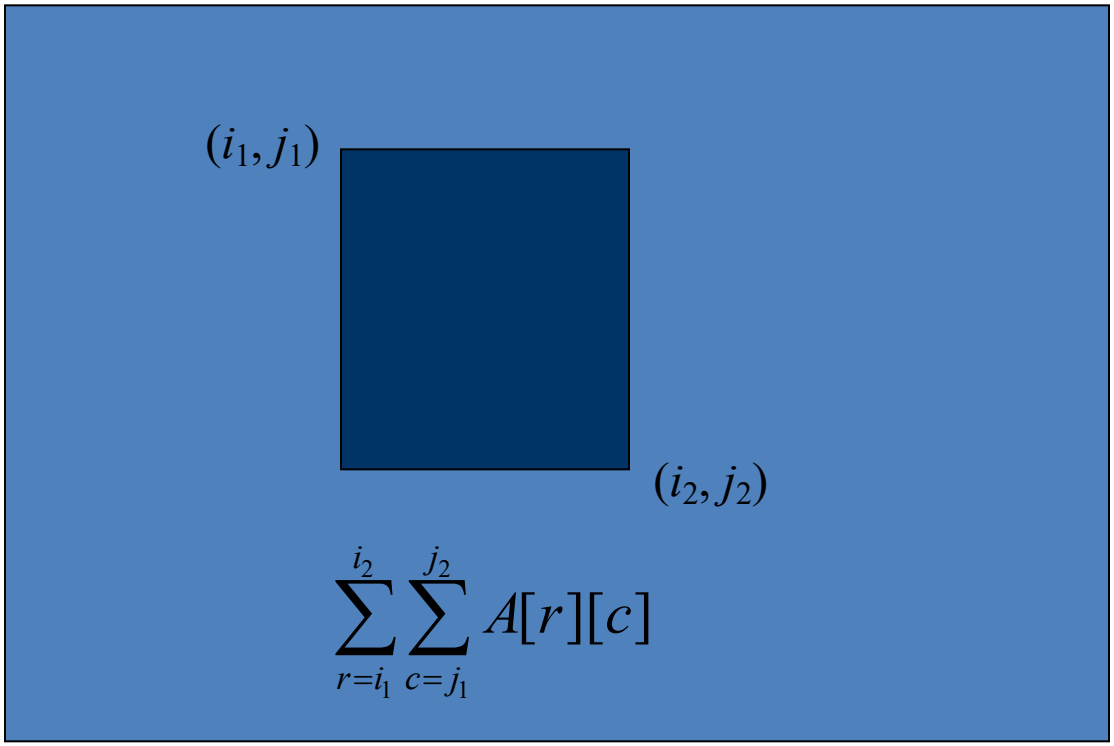
- Input (standard/console input)
  - testcase1000.txt

- Output (standard/console output)



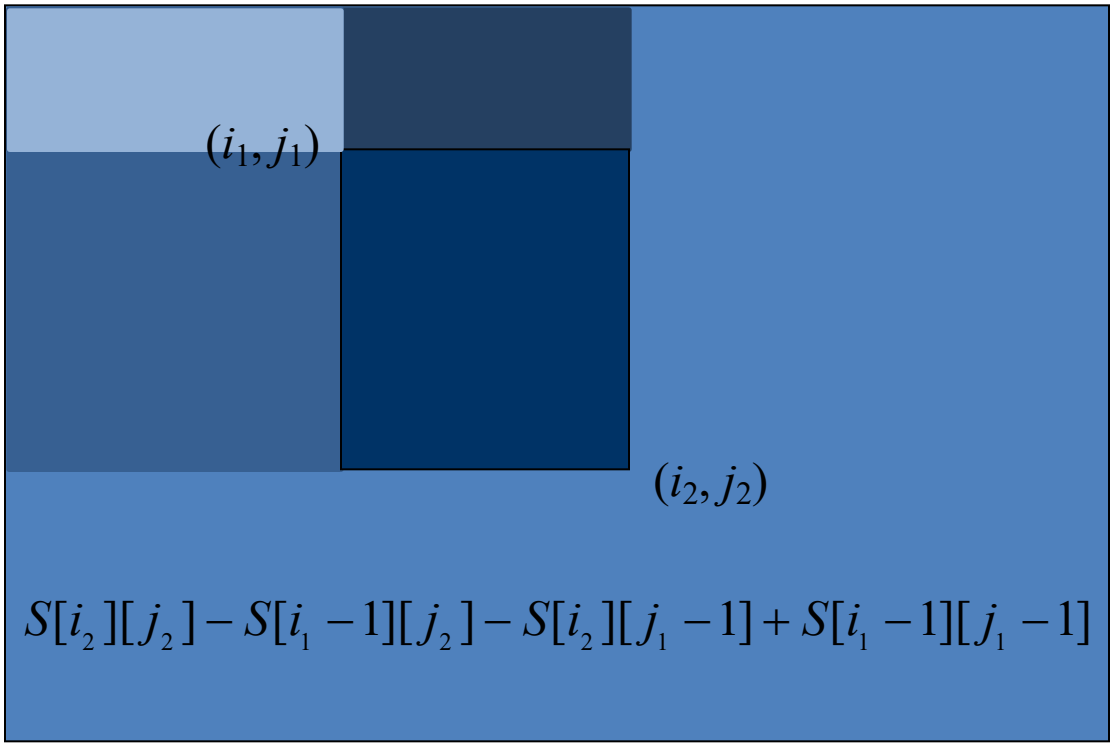
Naïve Method

- O(n^6)-time algorithm



Better Method

- O(n^4)-time algorithm



Your Method

- Design O(n^3)-time algorithm
  - Dynamic Programming

- Cf. Section 2.6 and Problem 2.f of the auxiliary textbook.

Due Date

- Soft deadline: **May 13, 2018**
- Hard deadline: May 20, 2018
  - But, will be deducted 10% per one day from your original score

Submission Date	Deduction Rate
May 14	10%
May 15	20%
May 16	30%
May 17	40%
May 18	50%
May 19	60%
May 20	70%
May 21	100%

Notice (cont'd)

- You should observe the format of input & output exactly.

- You should submit a compressed file (**HW2_your-ID.zip**) containing the following three files to the "u-campus" web-site (http://info.kw.ac.kr).
  - **HW2_your-ID.hwp/doc** // report document
  - **HW2_your-ID.cpp/cc** (or **.java**) // source code
  - **HW2_your-ID.exe** // executable file

Notice (cont'd)

- Source code
  - It should be compiled in
    - **C/C++ Language: Visual Studio 2010 or higher, or g++**
    - **Java Language: not restricted**
    - **You should note your environment in your report.**
  - Your name and student ID should be noted at the top of your source files in the form of comment.
- Report
  - Free format.
  - But, it must include several examples for testing your program and your own discussion.
  - It will be an important factor for getting a good score.
  - Mention your programming language together with compiler.