COP 3503C Programming Assignment # 6

Time Limit: 2 seconds (per input case)

NO LATE SUBMISSION FOR THIS ASSIGNMENT

Read all the pages before starting to write your code

What should you submit?

Write all the code in a single Main.java file. **Submit your** Main.java file **to Codegrade platform. Also, submit 2 test files (test_1.txt and test_2.txt) along with your test strategy test_strategy. In the test strategy, write your test strategy and discuss about the two test files you have generated and what is the correct output for those test files.**

Please include the following commented lines in the beginning of your code to declare your authorship of the code:

/* COP 3503C Assignment 6

This program is written by: Your Full Name */

Compliance with Rules: UCF Golden rules apply towards this assignment and submission. Assignment rules mentioned in syllabus, are also applied in this submission. The TA and Instructor can call any students for explaining any part of the code in order to better assess your authorship and for further clarification if needed.

Caution!!!

Sharing this assignment description (fully or partly) as well as your code (fully or partly) to anyone/anywhere is a violation of the policy. I may report to office of student conduct and an investigation can easily trace the student who shared/posted it. Also, getting a part of code from anywhere will be considered as cheating.

Deadline:

See the deadline in Webcourses. ***NO LATE SUBMISSION FOR THIS ASSIGNMENT***. An assignment submitted by email will not be graded and such emails will not be replied according to the course policy.

What to do if you need clarification on the problem?

I will create a discussion thread in webcourses and I highly encourage you to ask your question in the discussion board. Maybe many students might have same question like you. Also, other students can reply and you might get your answer faster. Also, you can write an email to the TAs and put the course teacher in the cc for clarification on the requirements.

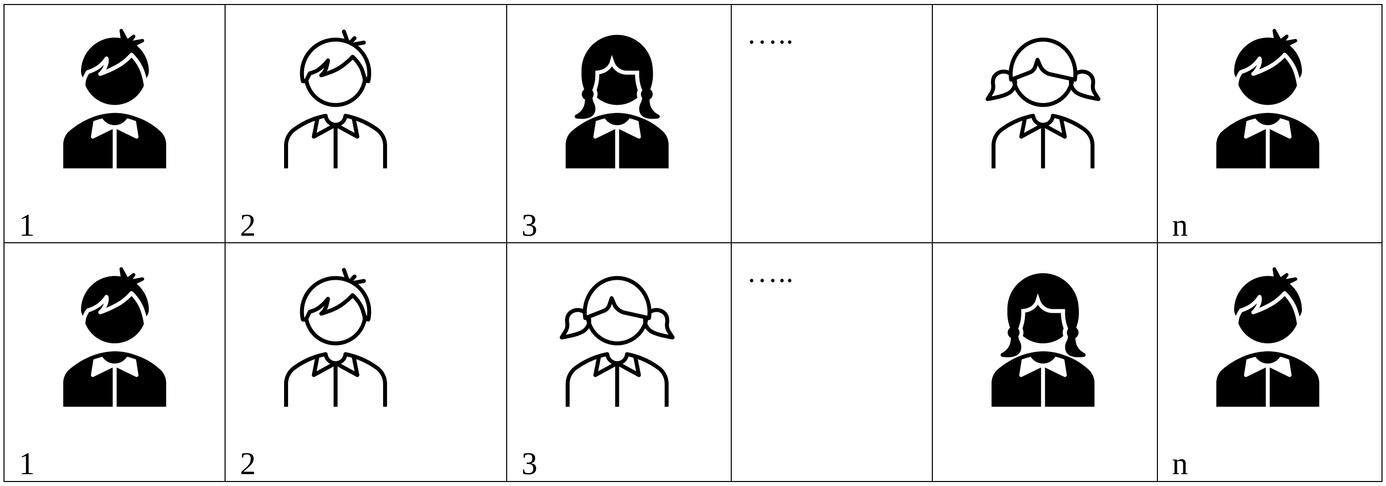
How to get help if you are stuck?

According to the course policy, all the helps should be taken during office hours. Occasionally, we might reply in email.

Problem Description: Team Building

An instructor teaches two CS2 sections (section 1 and section 2). He is thinking to short list a group of students for an upcoming contest. As part of the process, n number of candidates from each section has solved several programming problems.

On the day of the selection, 2n candidates have come to the venue for short listing. He asked the students to stand in 2 rows of same size. Exactly n students from each section. The candidates are numbered from 1 to n on each row from left to right.



1	2	3	…..		n
1	2	3	…..		n

Now the instructor will choose the candidates from the rows with the following rules:

1. Candidates should be chosen from left to right.
2. Index of each chosen candidate (excluding the first one taken) will be greater than the index of the previously chosen candidate.
3. In order to avoid giving preference of a specific section, he will choose the candidates in such a way that no consecutive chosen candidates belong to the same section (aka row).
4. The first student can be chosen from all 2n students without any constraints.
5. The final resulting group can have any number of candidates.
6. In order to make a perfect group of students, he will choose the students in such a way, that the total number of problems solved by the students will be the maximum.

So, solve this problem to find the group of students that will maximize the problems solved with all the given constraints.

Input specification (Must be read from standard input (no file i/o))

The first line of the input contains a single integer n ($1 \le n \le 100000$) that represents the number of students in each row.

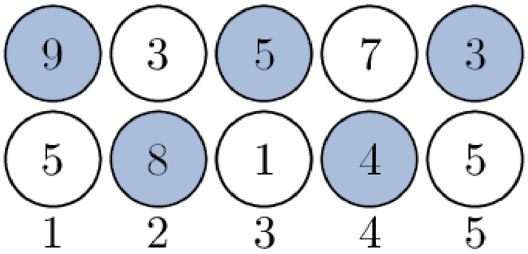
The second line of the input contains n integers $p_{1,1}, p_{1,2}, \ldots, p_{1,n}, (1 \le p_{1,i} \le 10^9)$, where $p_{1,i}$ is the number of problems solved by the ith student in the first row.

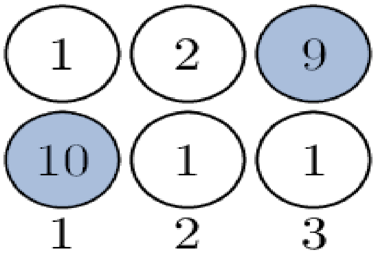
The third line of the input contains n integers $p_{2,1}, p_{2,2}, \ldots, p_{2,n}, (1 \le p_{2,i} \le 10^9)$, where $p_{2,i}$ is the number of problems solved by the ith student in the second row.

The Output (standard console output):

Print one number — the maximum possible total problem solved by the selected group of students.

Sample Input	Sample Output	Explanation

5
9 3 5 7 3
5 8 1 4 5
	29	The chosen candidates can be these:
		

3
1 2 9
10 1 1
	19	

1 7 4	7	

Some Restrictions:

- **You must have to use Dynamic Programming with bottom up tabulation approach to solve this problem**
- **The run-time has to be linear O(n)**

Hints:

- The code will be very small but require some good understanding of bottom-up tabulation approach of dynamic programming. So far, we have seen Fibonacci, 0/1 knapsack, LCS, MCM, and some other examples. These all should give you some ideas on dynamic programming approaches.
- As you now know dynamic programming, don’t start thinking on brute force approach.
  - Have a 2D table based on n and number of rows
  - Initialize the first index of row 0 and row 1

- Start updating index 1, row 0 and then then update index 1 or row 1, keep reading the hints to understand how to decide the numbers for the update.
- Remember, your target is to maximize the total.
- So, while updating an index of row 0 of the table, it should be based on some max of last calculations, and new calculations. Also note that, an update of row 0 in the table might need information from row 1. Same for row 1.
- You will generally no need to access information other than the previous index and the candiate list's current index
- Finally, your final answer will be located in the last index of row 0 or row 1 of the table.

What To Submit

See the first page of the assignment about what to submit.

Good Luck!