

**Rowan University**  
**Department of Computer Science**

**13<sup>th</sup> Annual Regional South Jersey High School Programming Contest**  
Saturday, March 27<sup>th</sup>, 1999

**Individual Contest**

***Problem description***

Consider the problem of locating a maximum sub-array in an array of integers, that is, a sub-array with the maximum sum. A maximum sub-array is defined by its boundaries, the indexes where it begins and ends, and there must be no other sub-array with a larger sum. For example, consider the array depicted below:

5	2	-1	-4	3	2	1	-9	12	3	17	-2	-7	11	-3	-4	15	-6
---	---	----	----	---	---	---	----	----	---	----	----	----	----	----	----	----	----

Its maximum sub-array starts ...            here ↑            ... and ends ...            here ↑.

(If you are programming in C++ or Visual Basic, it starts at index 8 and ends at index 16. If you are programming in Pascal, then the starting index is 9 and the ending index is 17. )

This maximum sub-array has a sum of 56 ( $12 + 3 + 17 + -2 + -7 + 11 + -3 + -4 + 15 = 56$ ).

Develop a program that locates a maximum sub-array in an array of integers. Your program should output the index (position) where the maximum sub-array begins, and the index (position) where the maximum sub-array ends, and the sum of the maximum sub-array. Your program's input and output should be exactly as illustrated below.

Your program must be general and work for any array of integers. For simplicity during the contest, you are allowed to make two assumptions: 1) that every array that your program will process contains at least one non-negative element, and 2) that the total number of elements in every array that you program will process is 100 or less.

Remember that the judges will use the following criteria to evaluate your program, in order of decreasing importance: Program output, readability and documentation, methodology, and programming structure (see the blue sheet in your folder for more details). The important notion of efficiency in this problem refers to the total number of array accesses made by your program.

***Input/output requirements***

The input values to be used are provided on a web page. If you call your array **a**, your program can initialize your array using the exact code on the web page. Obtain the input values by going to the web page for your programming language:

- For C++, <http://www.rowan.edu/~baliga/cpp>
- For Visual Basic, <http://www.rowan.edu/~baliga/vb>
- For Pascal, <http://www.rowan.edu/~baliga/pascal>

Your program should find 4 maximum sub-arrays:

1. The maximum sub-array in the first 25 elements of **a**.
2. The maximum sub-array in the first 50 elements of **a**.
3. The maximum sub-array in the first 75 elements of **a**.
4. The maximum sub-array in all the 100 elements of **a**.

Your program should produce the following output:

Contestant id *display-your-id-here*

The maximum sub-array in the first 25 numbers begins at index **B1**, ends at index **E1**, and has a sum of **S1**.

The maximum sub-array in the first 50 numbers begins at index **B2**, ends at index **E2**, and has a sum of **S2**.

The maximum sub-array in the first 75 numbers begins at index **B3**, ends at index **E3**, and has a sum of **S3**.

The maximum sub-array in the first 100 numbers begins at index **B4**, ends at index **E4**, and has a sum of **S4**.

Note: In your output, **B1**, **E1**, **S1**, **B2**, **E2**, **S2**, etc., will be substituted for the actual values computed by your program.

### ***Submission instructions***

Submit before the two and a half-hour deadline:

- 1) a diskette containing your program,
- 2) a printout of your program
- 3) a printout of your program's execution