|  |  |
| --- | --- |
| **Project 3** | **CMPT 438 Algorithms** |
|  | **Fall 2016** |

The maximum subarray problem can be solved in linear time! Use the following ideas to develop a **nonrecursive**, **linear-time** algorithm for the maximum-subarray problem.

Start at the left end of the array, and progress toward the right, keeping track of the maximum subarray seen so far. Knowing a maximum subarray of A[1 .. j], extend the answer to find a maximum subarray ending at index j+1 by using the following observation: a maximum subarray of A[1 .. j + 1] is either a maximum subarray of A[1 .. j] or a subarray A[i .. j + 1], for some 1 <= i <= j + 1. Determine a maximum subarray of the form A[i .. j + 1] in constant time based on knowing a maximum subarray ending at index j.

Submit a MS word document to Moodle (lms.manhattan.edu) that consists of the followings:

* Source code in either C++ or Java
* Screenshot of the output
* A brief explanation of why your program has the running time of O(n)