School of Computer Science, University of Windsor

60-254: Data Structures and Algorithms

Term: Summer 2021

Instructor: Dr. Asish Mukhopadhyay

Lab 2
Posted: 21 May, 2021
Due: Due 27 May, 2021

Instructions:

- You are expected to finish the lab by the end of the posted date. Submissions beyond the due date will earn a penalty of $n * 25\%$, where $n = submissionDay - dueDay$. Thus if the lab is due Tuesday and you submit on Wednesday, this will be considered a day late.
- You will have to upload your work as a script file via the BLACKBOARD portal for record-keeping and off-line grading. Email submissions are not accepted. Create a script file as follows:

```
1. script labName.txt
2. cat labName.c
3. cat input.txt (ignore this line if no input.txt file has been created)
4. cc labName.c
5. ./a.out < input.txt (./a.out, if there is no input.txt file)
6. ls -l
7. exit (DO NOT FORGET THIS STEP!!)
```

  Step 3. is necessary if you are creating a data file, input.txt, to read the input from.

- There will be no make-up for missed labs. If you have missed a lab for truly extenuating circumstances (like illness or family emergency) I will consider allowing you to make a late submission. However, I need to be informed by email about this on the day of the missed lab. The email should include your name and SID.

Problem:

In class we discussed two different algorithms for the maximum subsequence problem (one brute-force; the other more sophisticated that runs in $O(n)$ time).

Here's a third algorithm. For each element a_i of the sequence, $a_0, a_1, \ldots, a_{n-1}$ (notice that the indexing starts from 0 to be consistent with the indexing convention in C), compute the start index and the value of the largest contiguous subsequence (segment is simpler and which is what I shall use) that ends at a_i. This can be done in one traversal from the left to the right. At the same time, record the starting index of this maximum segment sum, and the end index i (obviously!).

Once this is done, in another left-to-right scan one can determine the largest segment sum, and the start and end indices of this segment.

The following example for a sequence of 10 integers might help in understanding the problem better.

index	0	1	2	3	4	5	6	7	8	9
sequence	-99	94	-96	12	99	41	-4	-62	-29	50
start index	-1	1	-1	3	3	3	3	3	3	3
segment sum	0	94	0	12	111	152	148	86	57	107

Let us explain the entries in the 5th column. The index value is 4, and we have an entry 99 below this as $a_4 = 99$. In the box below, we initially enter 4 and in the box below that 99, as we have a candidate solution for the largest contiguous sequence starting and ending at index 4, with value 99. However, the largest contiguous sequence ending at index position 3 is 12 and we notice that we can extend this to index position 4 to get a value 111 (99 + 12) which is larger than 99 and starts at 3. Hence the entries 3 and 111 in the 3rd and 4th rows respectively of the 5th column.

For a set of 10 randomly generated sequences, each consisting of 15 integers, print the value of a maximum subsequence that you obtain, as well as the start-index and the end-index of this maximum subsequence. There can be several; it's enough to report one. (10 points)