

**York University**  
**Lassonde School of Engineering**  
**Dept. of Electrical Engineering and Computer Science**  
**EECS 2032**  
**Embedded Systems**  
**Winter 2020**

---

<b>EECS2032Z</b>	<b>Lab Test 2_A</b>	<b>Intro to Embedded Systems</b>
------------------	---------------------	----------------------------------

---

Thursday, March. 12<sup>th</sup> 2019

4:00– 5:50pm

Question 1 (4 points)

Write a C program to read one integer and print the integer that is formed by reversing the order of digits in a default format (%d). For example, if the input is 65160, the output should be 6156. It continues to do this until it reads a zero

For example (Blue is user input, red is computer response)

**CPU % a.out**

**123456**

**654321**

**17**

**71**

**0**

**CPU%**

**submit 2032Z      labtest2A   labtest2\_1A.c**

Question 2 (5 points)

Write a C program to read an integer ( $n$ ) followed by  $n$  integers (vector of length  $n$ ). Your program should display the maximum two elements of the vector and their positions in the vector. The result should be displayed on 2 lines, where the first line is the maximum number, followed by a tab, followed by its position in the array (positions start with 0). Then on the second line display the second largest element, followed by a tab, followed by its position and then new line. In case of two elements with the same value, display the one with the smaller position. Consider the maximum value of  $n$  is 25. Also consider the first element of the array is in position 0

(System prompt and response in red, user input is blue)

**CPU % a.out**

7

1 6 8 3 2 5 23 3

23 6

8 2

**CPU%**

**submit 2032Z labtest2A labtest2\_2A.c**

Question 3 (6 points)

Write a C program to read two integers ( $r, c$ ) followed by an  $r$ -by- $c$  matrix of integers. The program should check every column if the column is sorted in an ascending order, descending order, or not sorted at all). For every column the output should be either 'A' for ascending, 'D' for descending, or 'N' for not sorted. The output for the whole program is the output for each column separated by tabs. Maximum size of the array is  $15 \times 15$

For example (system prompt in red, user input in blue)

**CPU % a.out**

3 6

1 8 6 9 0 9

2 9 1 8 2 5

5 2 1 7 4 9

A N N D A N

**CPU%**

**submit 2032Z labtest2A labtest2\_3A.c**