**Full name: ID**

**Lab 9:**

Give the following an array of integers in MIPS:

1. Suppose that the array *Array* is zero-terminated. Write an assembly procedure len(Array) using stack that returns the length of the array (20points). Run it step by step and highlight the address, value of stack.

.data

Array: .float 43.01, 34.06, XY.09, 78.9, 64.98, 2126.1, 643.2, 451.4 ,423.4, 45.5, 566, 0

***Where XY is the whole digit of your ID, e.g. ITITIU22016 🡪 XY is 22016***

2. Write sort(Array) procedure that prints out the ordered array using Quick sort (40points)

Hint: [https://www.programiz.com/dsa/bubble-sort](https://colab.research.google.com/corgiredirector?site=https%3A%2F%2Fwww.programiz.com%2Fdsa%2Fbubble-sort)

intArray: .word 43, 6543, 34, 54, 4232, 64, 526, 643, 6435 ,423, 4236, 566, **XY,** 0

**where XY is the last of two digits of your ID**

Please compare this sort with Bubble-sort

Run it step by step and highlight the address, value of calling Sort.

3. Write a secondMax(Array) that \_nds second highest value in the array. This procedure

should call the sort(Array) procedure in Exercise 2. Hint: you can use any sort to find this value (15points)

intArray: .word 43, 6543, 34, 54, 4232, 64, 526, 643, 6435 ,423, 4236, 566, **XY,** 0

**where XY is the last of two digits of your ID**

4. Write an assembly procedure normalized(Array) to normalize array and print it out (before and after normalize) (25points). Hint: divide the whole array with the maximum Array.

intArray: .word 43, 6543, 34, 54, 4232, 64, 526, 643, 6435 ,423, 4236, 566, **XY,** 0

**where XY is the last of two digits of your ID**