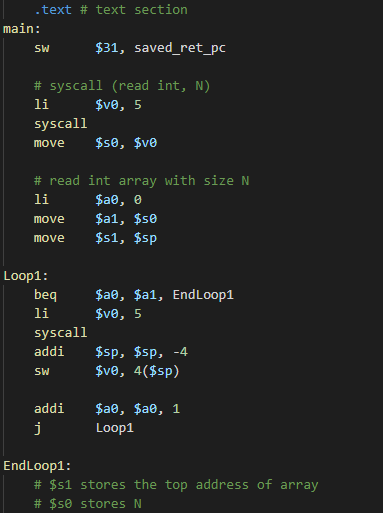
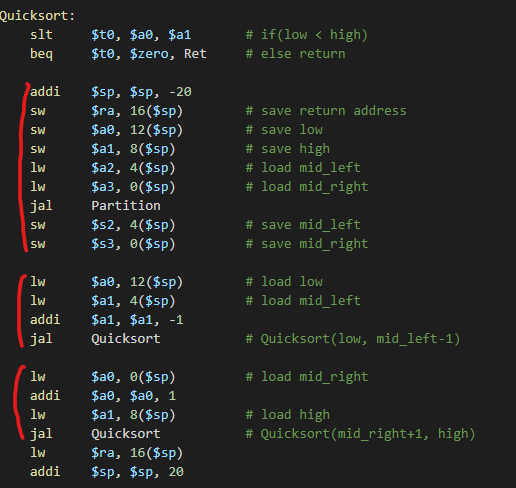
**HW2 Report**

**20200639 채우진**

1. **Stack Allocation**
   1. Getting series of integer size N:  
        
       Starting with the top address of the stack pointer, the received input was stored in each location while reducing the stack pointer. After I finished storing inputs, I stored the number of inputs, and the top address of array in $s0, $s1 respectively.
   2. Quicksort Implementation:  
        
      Everytime when I enter Quicksort, I decreased stack pointer by 20, to store 5 parameters; return address, low, high, mid\_left, mid\_right respectively. After each innated function(Partition, Quicksort, Quicksort) returned, I could reload modified value and use it as parameter for following functions.
2. **Implementation:**

The trick that I used for this programming assignment is “using interger array as a global variable” by storing it in $s1. It is quite efficient because I can save the number of arguments needed for calling functions. Specifically, the function ‘Partition' requires 5 parameters, although there are only 4 registers that I can use for arguments. If argument needed for function exceeds 4, then we should store extra arguments in the stack, but implementing those things are quite tough for me. So, I used the array as an global variable by storing it in the $s1.

