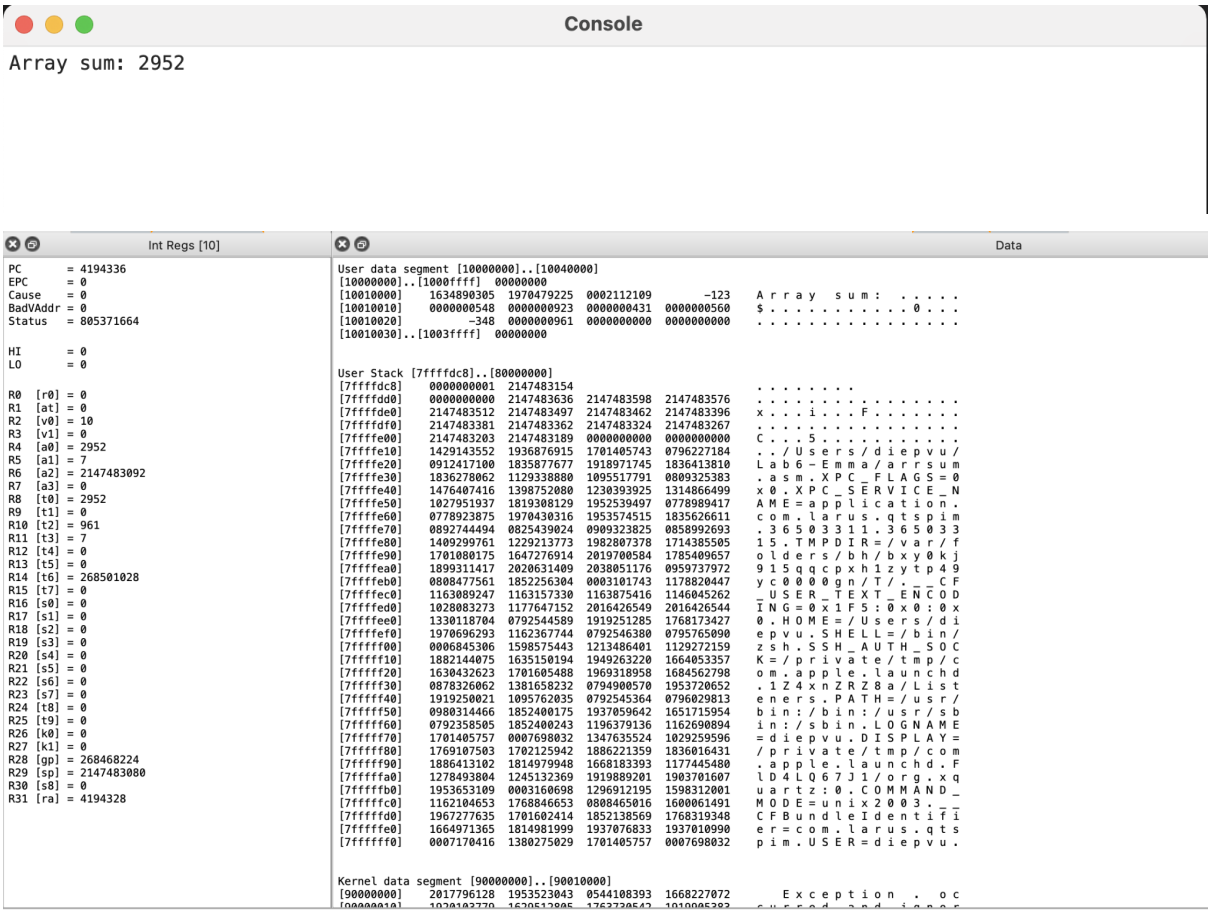
Diep (Emma) Vu

Lab 6 Report

Description of `arrsum.asm`

The main program calls subroutine `arrsum.asm` to get the sum of the array. The subroutine first checks if the array size is 0 to exit the loop since there is an initial jump before entering the loop. This works better for the condition of array size of 0 since it saves a jump into the loop.

Output for `arrsum.asm`



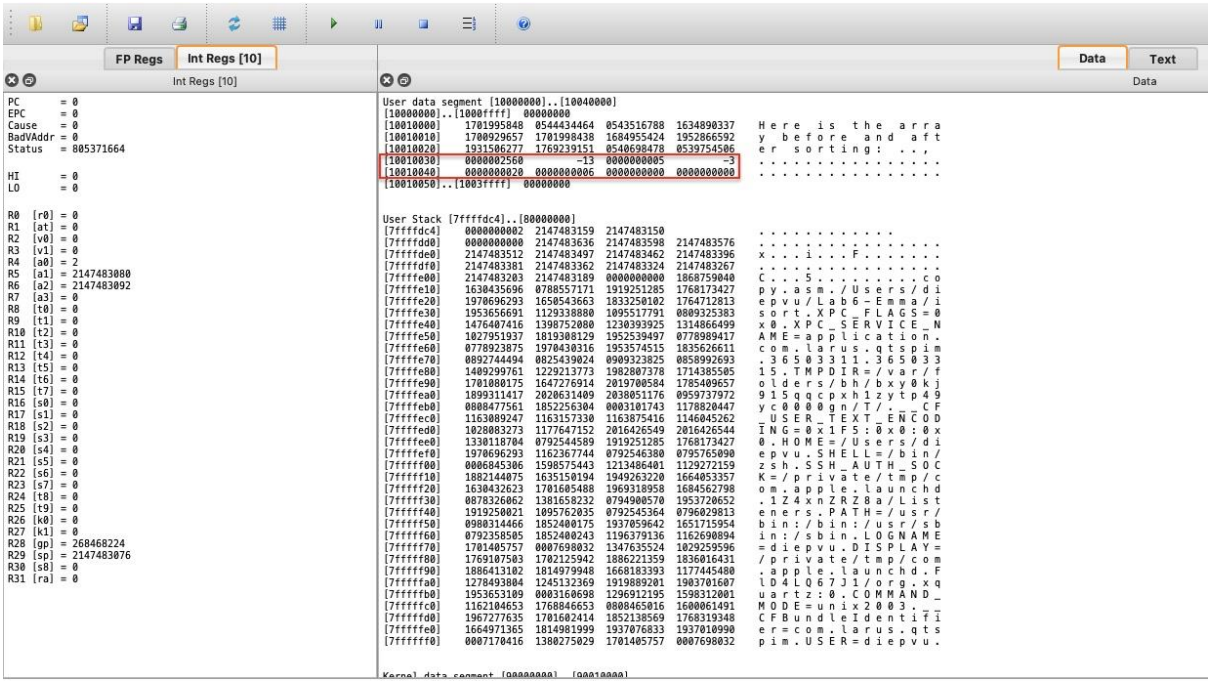
Description of `isort.asm`

The main program calls subroutine `isort.asm` to do insertion sort to an array. There's a pointer *i* and *j* saved in register *$t0, $t1*. Pointer *i* will go through the outer_loop starting from 1 and pointer *j* will go through the inner_loop starting from *i.*

I also had to extend the stack and preserve $a0 because it gets overwritten many times in order to call print_array subroutine before and after isort

Output for `isort.asm`

Before sorting:



After sorting: