HACETTEPE UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING BBM234



Name : Mehmet Taha

Surname : Usta

Number : 21527472

E~mail : [b21527472@cs.hacettepe.edu.tr](mailto:b21527472@cs.hacettepe.edu.tr)

Subject : Learning how to write and simulate MIPS code

Programming Language : MIPS ASSEMBLY

**1.Explanation of the problem**

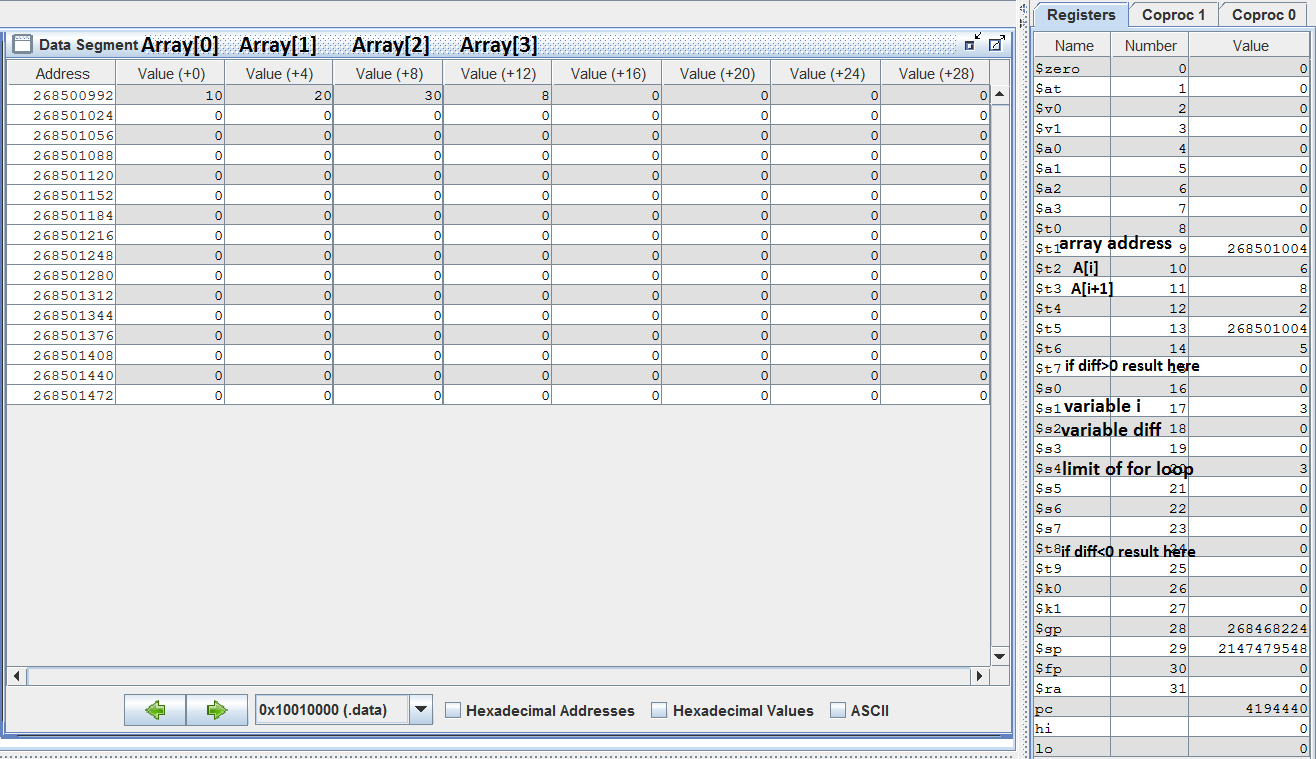
Define the problem array and reach the elements in the array in the first question. Then create a for loop to access the elements of the array. Obtaining results using if-else statement. Multiplication instructions are prohibited, so the multiplication instructions must define with the codes

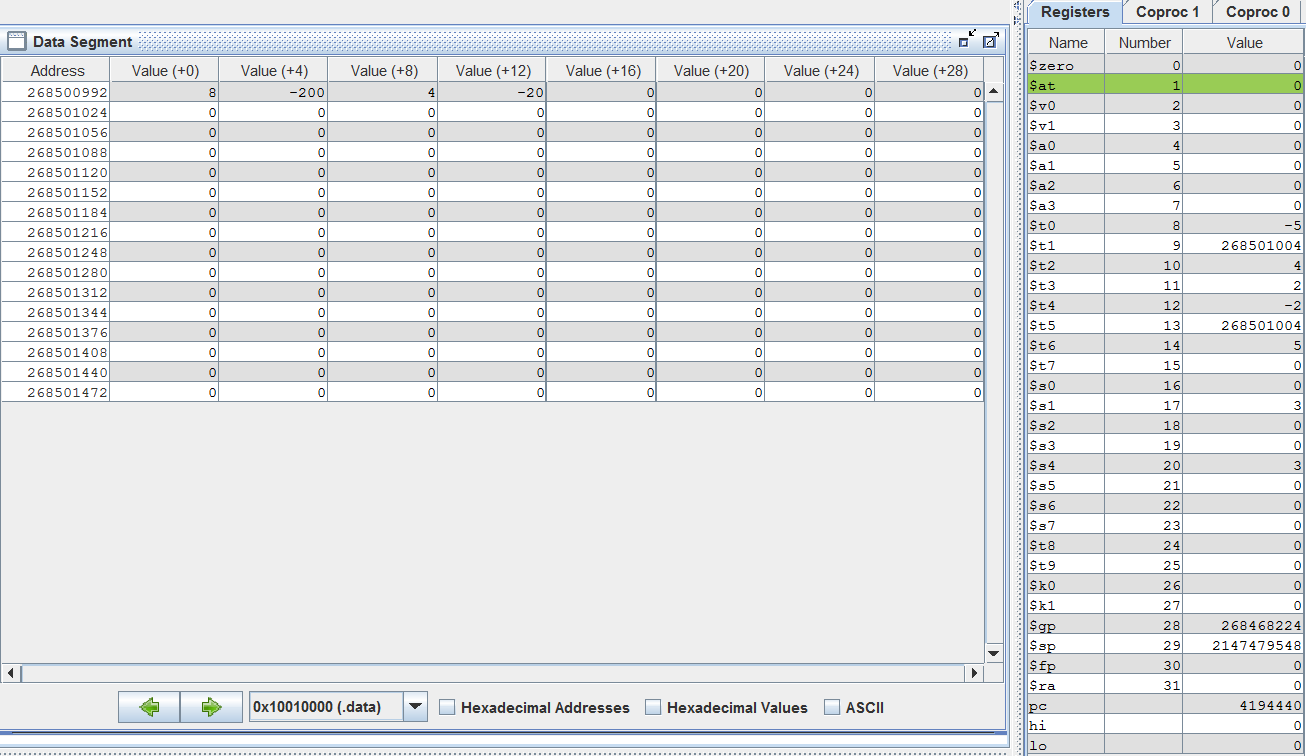
The second problem is that the defined functions are mutually connected

create and use stack structure to prevent loss of $ra between functions

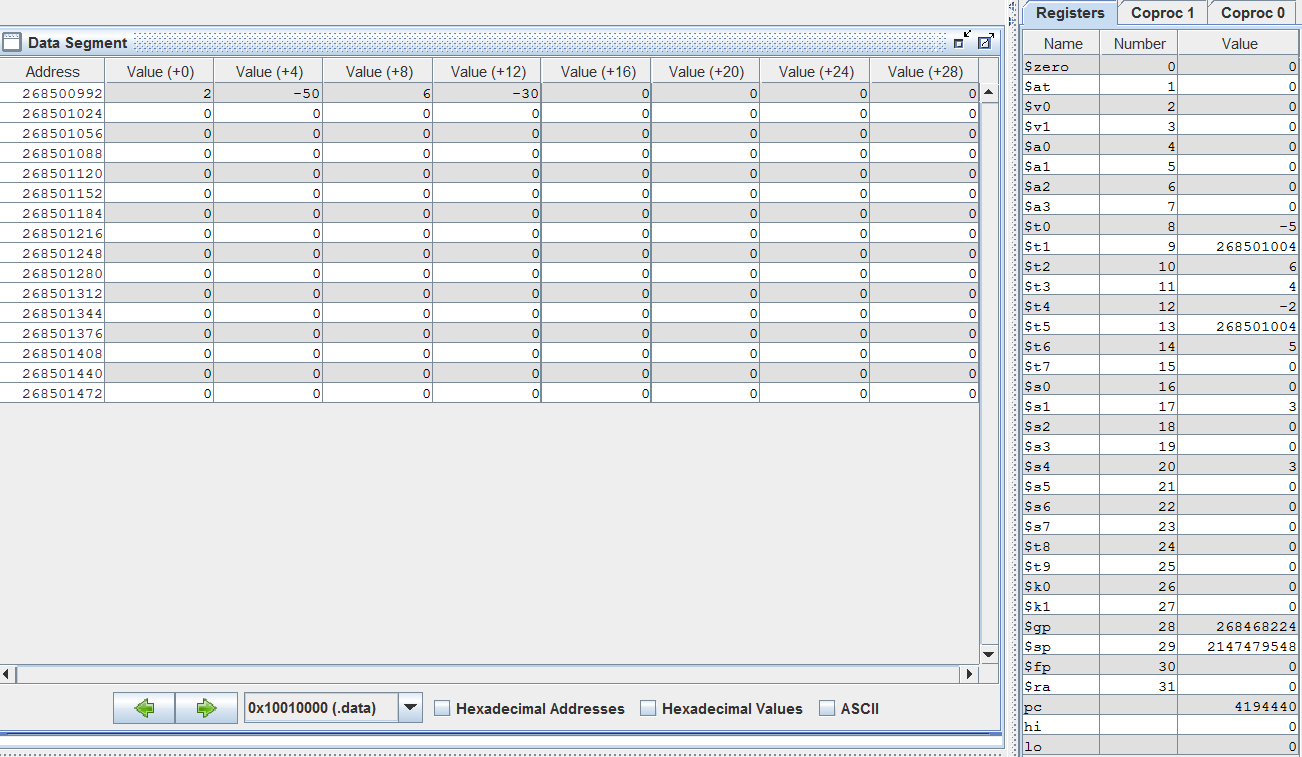
**2.Results**

**2.1)Array results**

Test 1: A={2,4,6,8} Test 2: A={8,6,4,2}

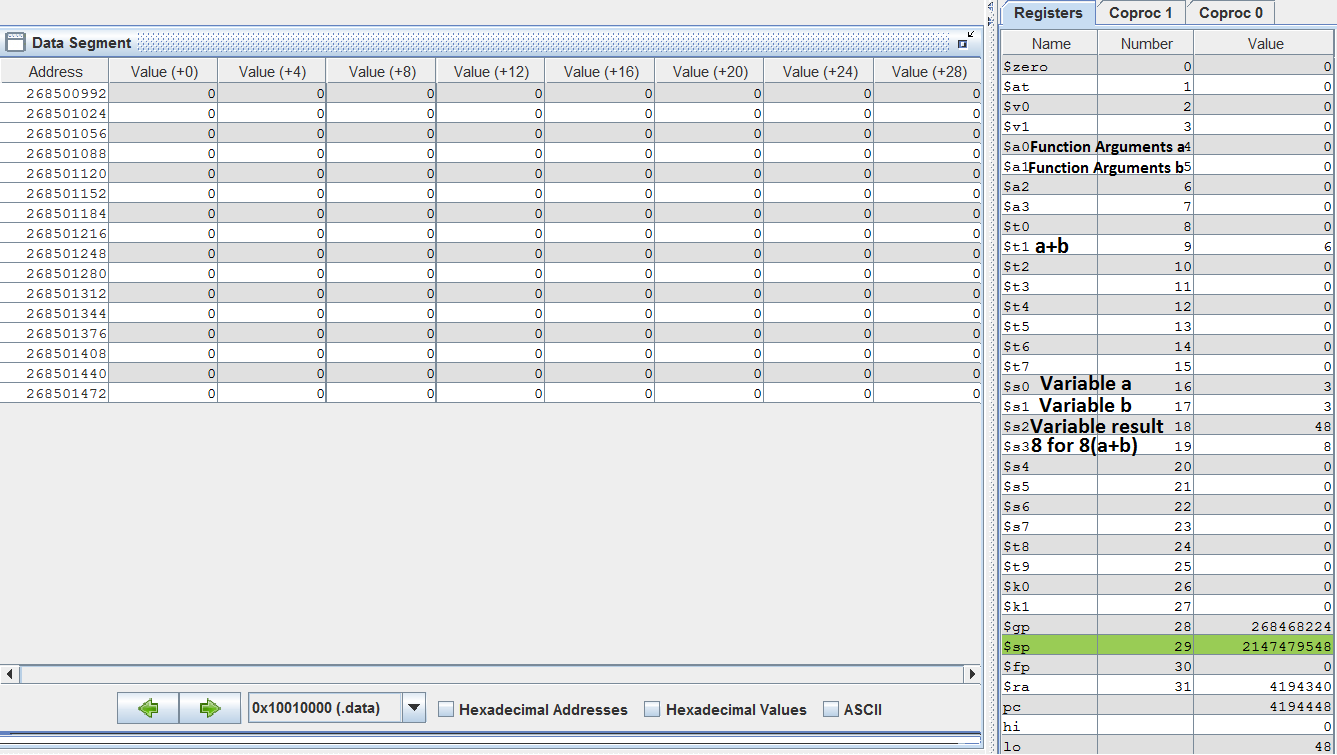


Test 3: A={2,2,6,4}



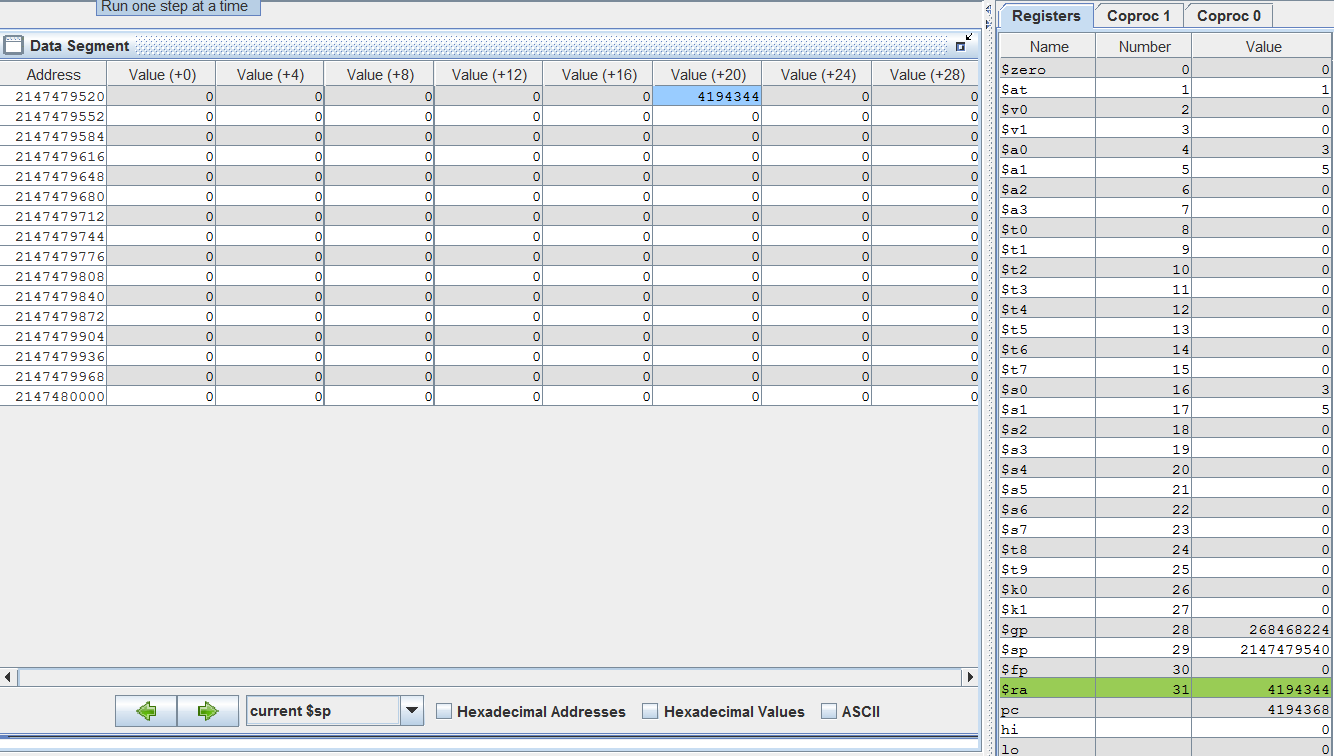
**2.2)Function calls results**

Test 1: a=3, b=3

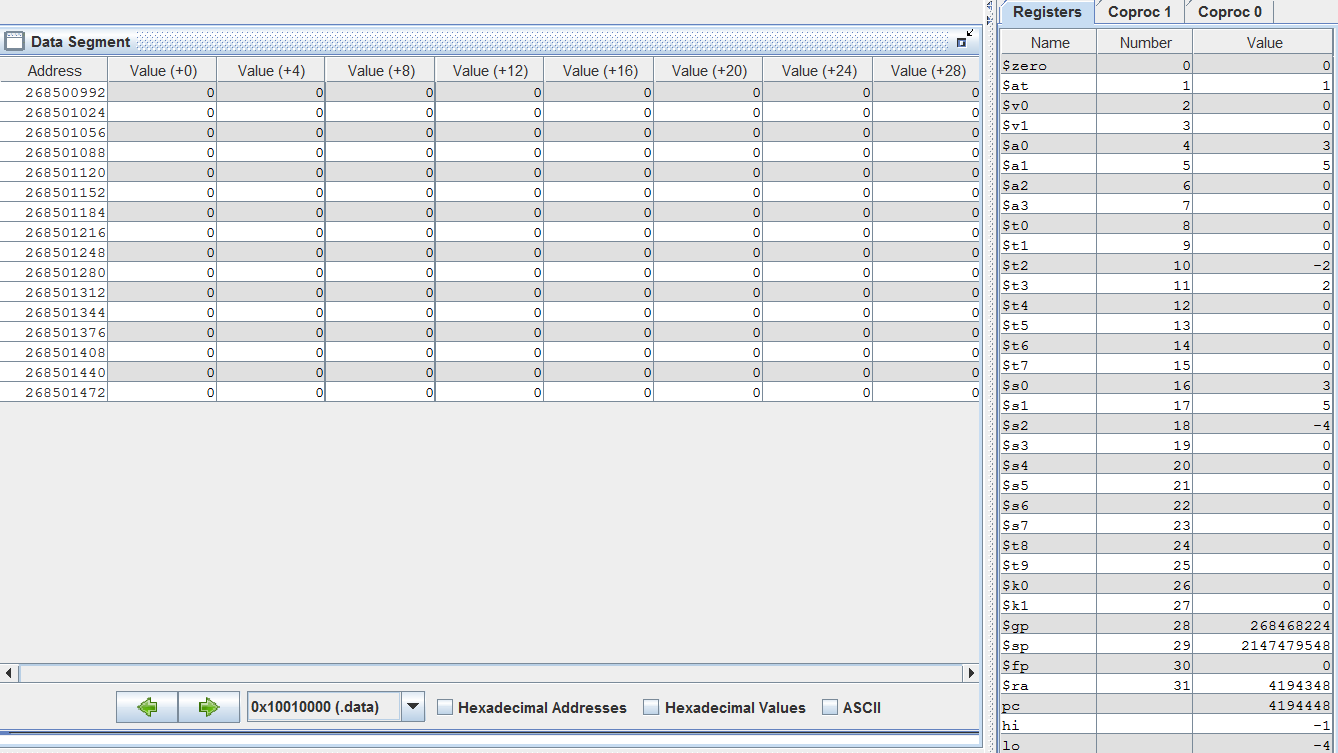


Test 2: a=3, b=5

When the return address($ra) stack pointer is stored

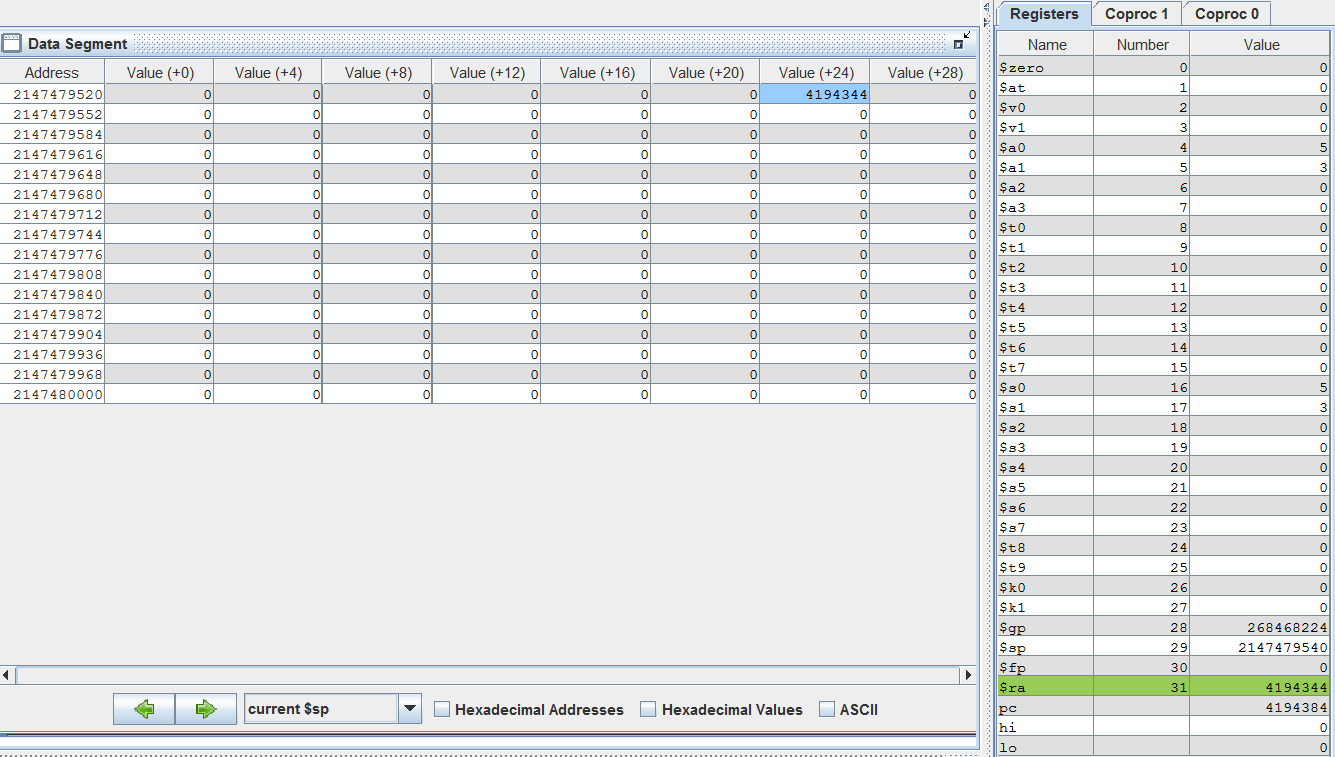


when the program is finished

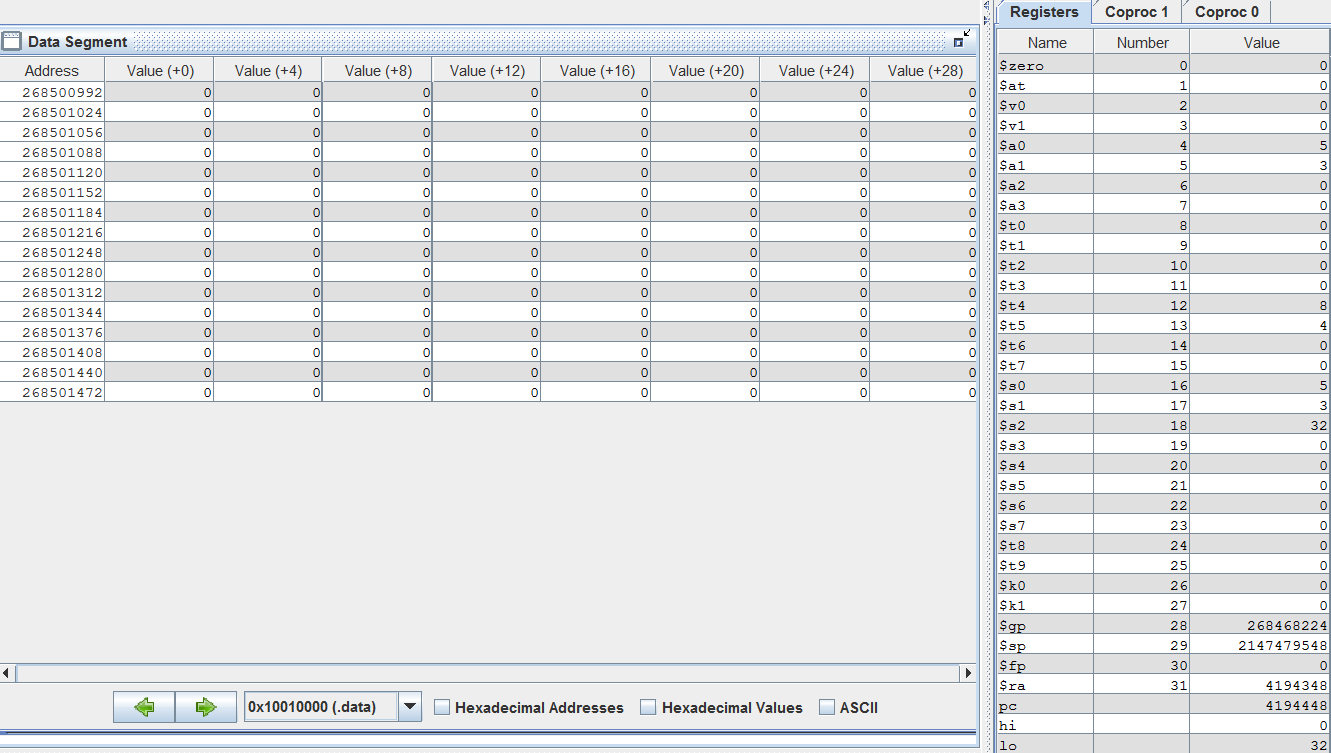


Test 3: a=5, b=3

When the return address($ra) stack pointer is stored



when the program is finished



**3)Codes**

**3.1)Array codes**

Codes first define array then define i and diff. for loop starts. A [i + 1] is subtracted from A [i] and diff is equal to this result. if diff is greater than 0 then A[i] = 5\*A[i]. if diff is less than 0 then A[i+1] =- 5\*A[i]. Multiplication is prohibited. defined 2 for loops in 2 different conditions(if-else) to describe multiplication

**3.2)Function Calls codes**

firstly I defined variables in main function. Then stack is allocated. I created if-else condition. if a and b are equal, the result is equal to 8 \* (a + b). if not equal, else will call compare function. The compare function directs incoming function arguments as punish and award. if a is smaller than b , punish function will work. if a is not smaller than b, award function will work. The punish and award functions work after the $ra(return address) is stored in the stack. the stack is used to remember the return address in interconnected functions .After mathematical operations are done,the result will be saved to Mehmet