U18CO018

Shubham Shekhaliya

Assignment – 6

MIT

1-> WAP to find Factorial of a given number using Call and Subroutine.

**Code :-**

**; for finding factorial**

**lxi h,4000H**

**mov b,m**

**call factorial ; calling first time**

**inx h**

**mov m,d ; store ans**

**inx h**

**mov b,m**

**call factorial ; calling second time**

**inx h**

**mov m,d**

**hlt**

**;after completion of this register d contains answer**

**factorial: mvi d,01H**

**next: call multiply**

**dcr b**

**jnz next**

**ret**

**;Helper subroutine**

**multiply: mov c,b**

**mvi a,00H**

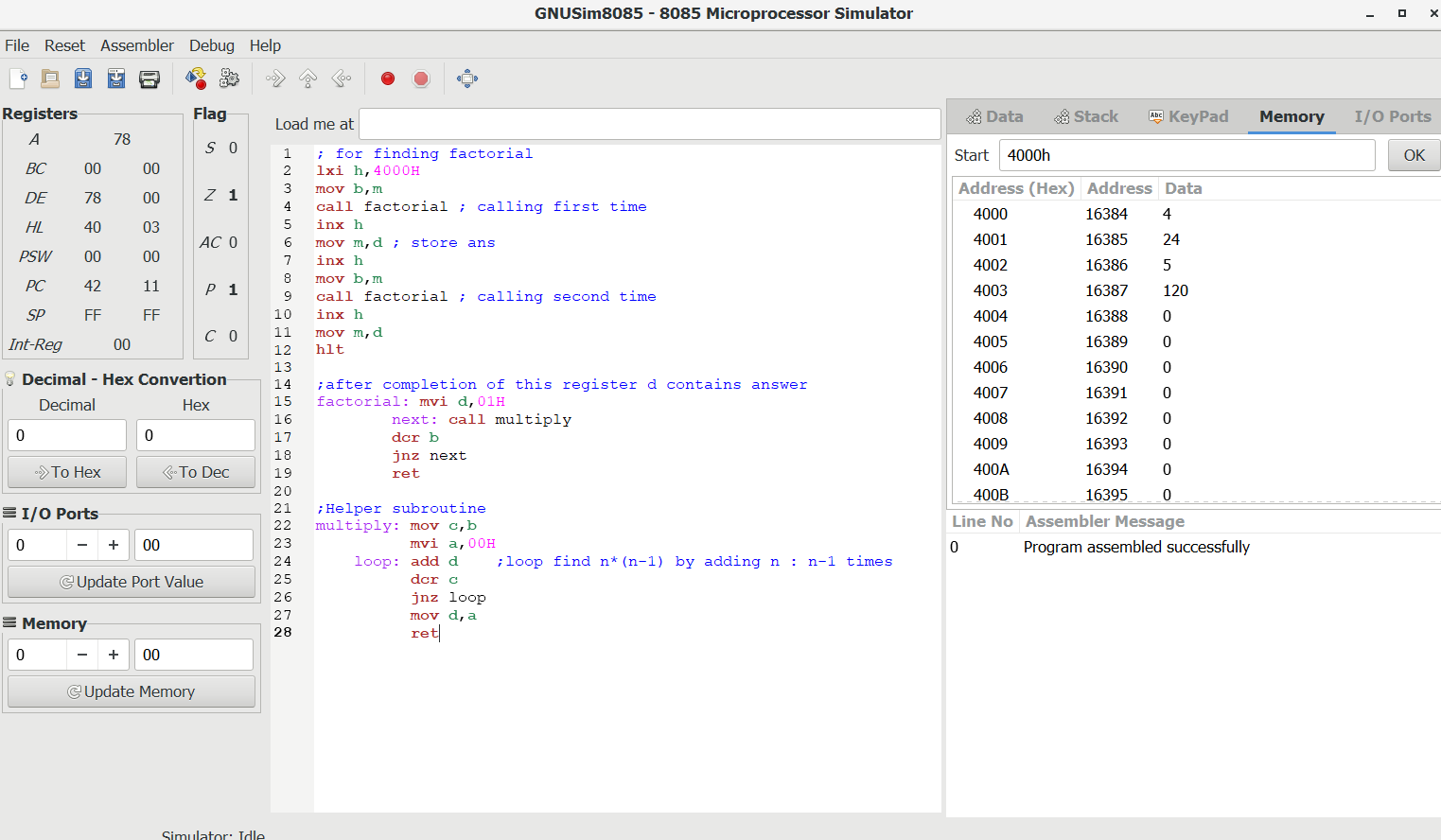
**loop: add d ;loop find n\*(n-1) by adding n : n-1 times**

**dcr c**

**jnz loop**

**mov d,a**

**ret**

****

2-> WAP for Fibonacci Series using Call and Subroutine.

**Code : -**

**; fibonacci series store**

**lxi h,4000H**

**mvi c,04H ;Counter how many we need**

**call fibonacci ;store address by h onwards**

**lxi h,4007h**

**mvi c,07H**

**call fibonacci**

**hlt**

**fibonacci: mov e,c**

**mvi b,00H ;base case 0th element**

**mvi d,01H ;base case 1st element**

**mov m,b**

**inx h**

**mov m ,d**

**dcr e**

**loop: call nextterm**

**dcr e**

**jnz loop**

**ret**

**;Calculate next item**

**nextterm: mov a,b**

**add d**

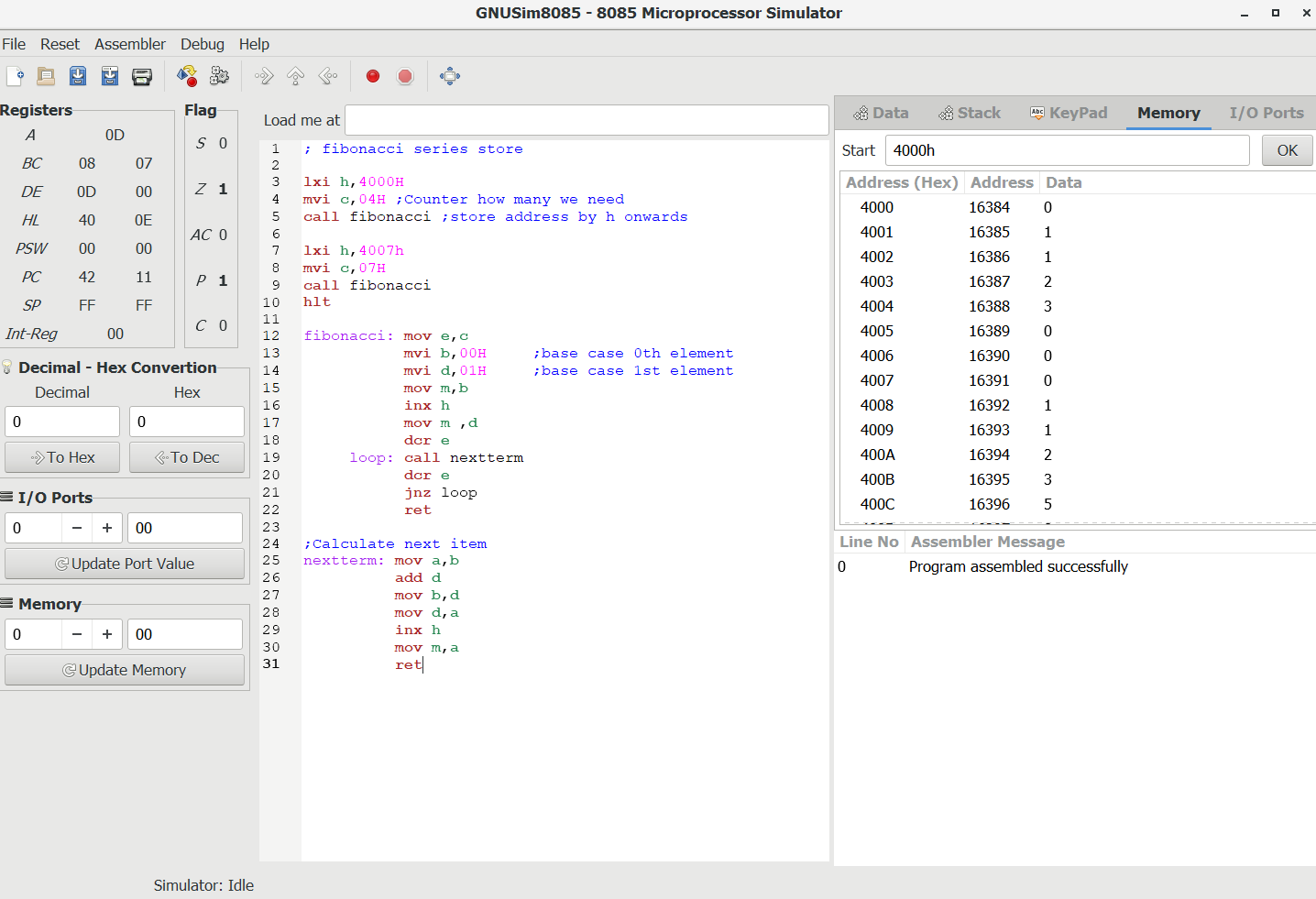
**mov b,d**

**mov d,a**

**inx h**

**mov m,a**

**ret**

****

3-> WAP to find Multiplication of Two 8-Bit Numbers using Call and

Subroutine.

**Code : -**

**lxi h,2000h**

**mov b,m ;get data**

**inx h**

**mov c,m ; get data**

**call multiply ; calling first time**

**inx h**

**mov m,a ;storing result**

**inx h**

**mov b,m**

**inx h**

**mov c,m**

**call multiply ;calling second time**

**inx h**

**mov m,a**

**hlt**

**multiply: mvi a,00H ;clear accumulator**

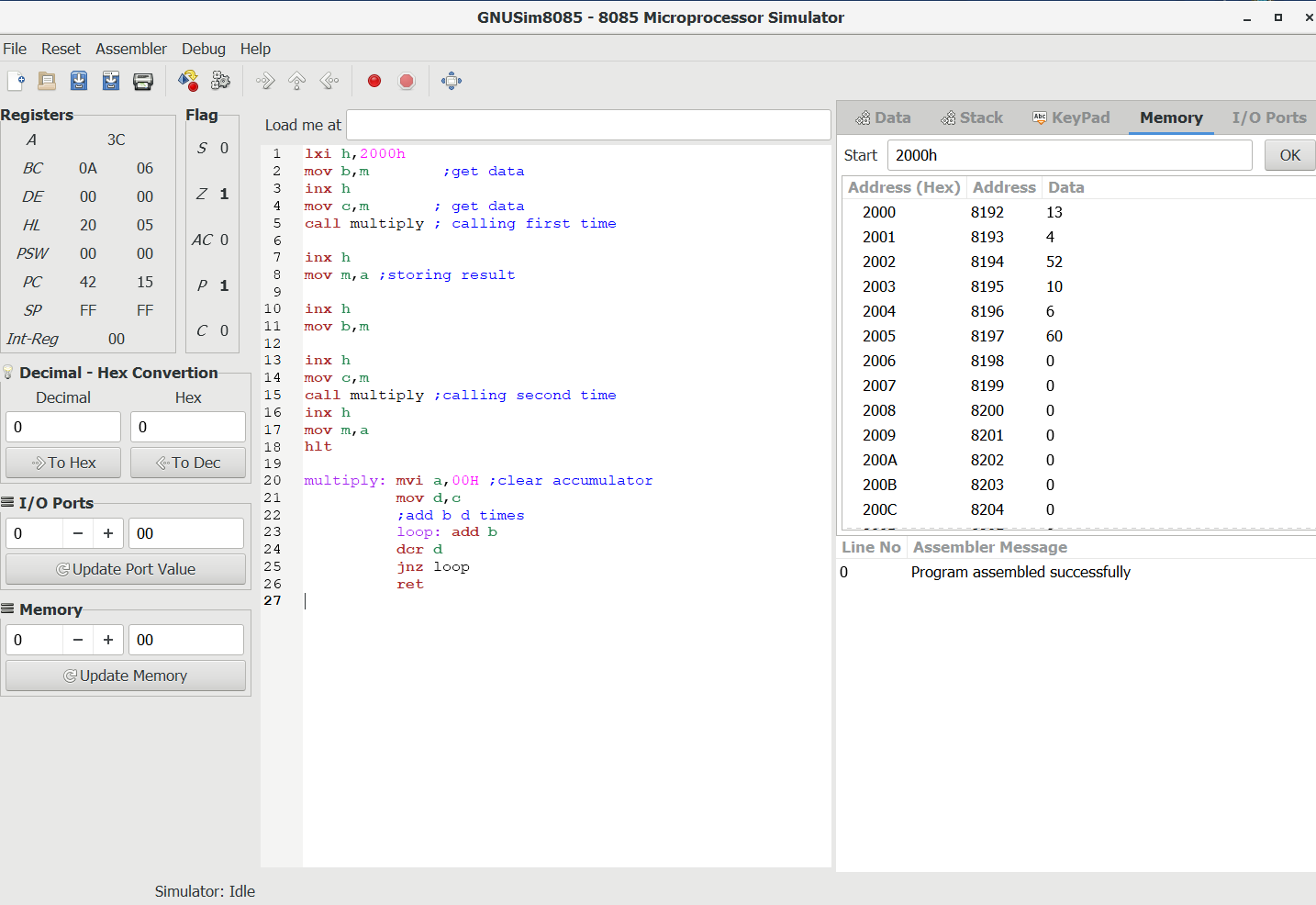
**mov d,c ;add b d times**

**loop: add b**

**dcr d**

**jnz loop**

**ret**

****

4-> Write Assembly language program to find the square/square root of a

number .The number is stored at location 5000H, store the result at 5050H.

**Code :-**

**lxi h,5000H**

**mov e,m**

**call squareroot**

**lxi h,5050H**

**mov m,d**

**lxi h,5001H**

**mov e,m**

**call squareroot**

**lxi h,5051H**

**mov m,d**

**hlt**

**; square sub routine which is used by square root also**

**square: mov c,b**

**mvi a,00H**

**loop: add b**

**dcr c**

**jnz loop**

**ret**

**; it find floor value of sqare root**

**squareroot: mvi d,00H**

**loop2: inr d**

**mov b,d**

**call square ; to check whether d\*d <= e**

**cmp e**

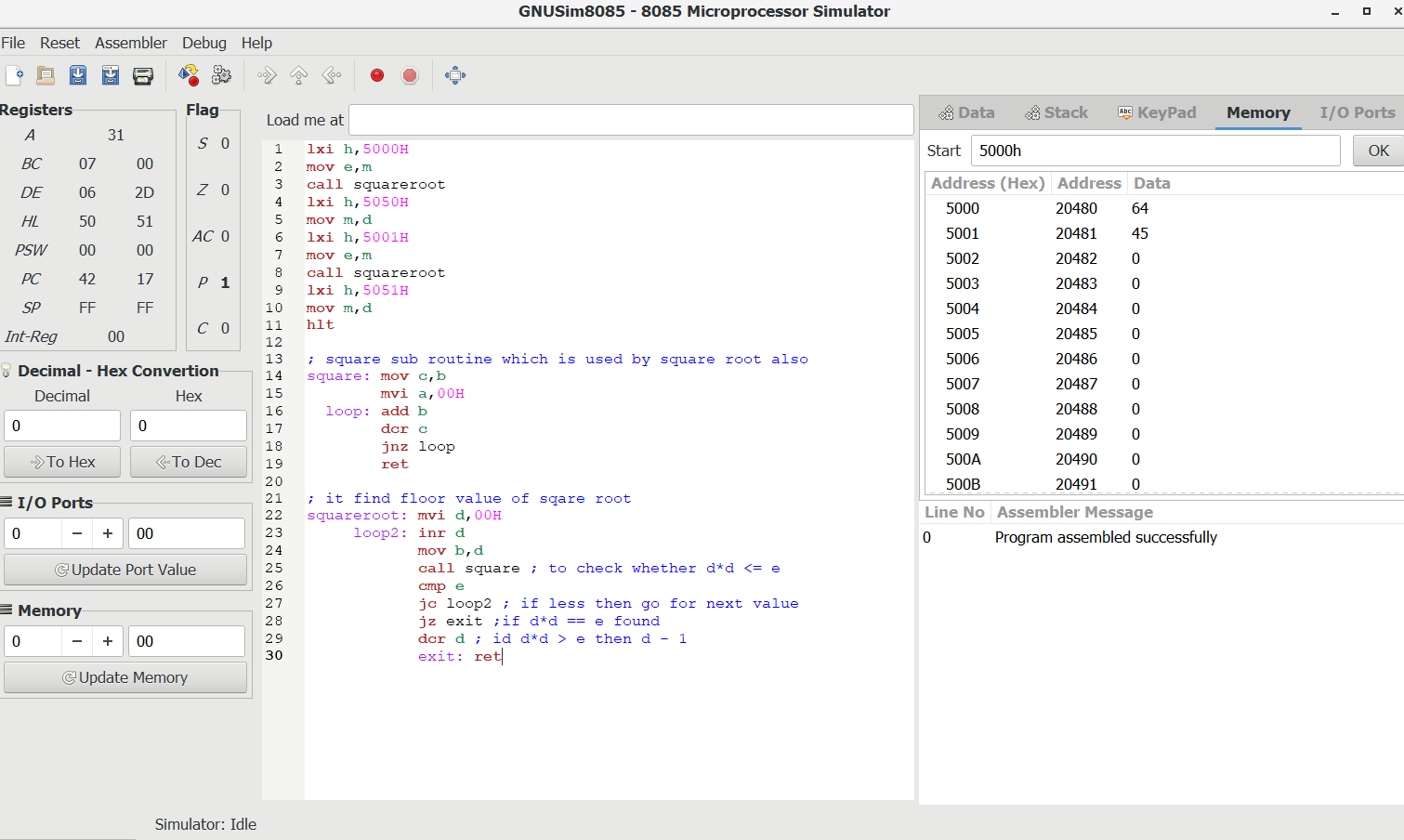
**jc loop2 ; if less then go for next value**

**jz exit ;if d\*d == e found**

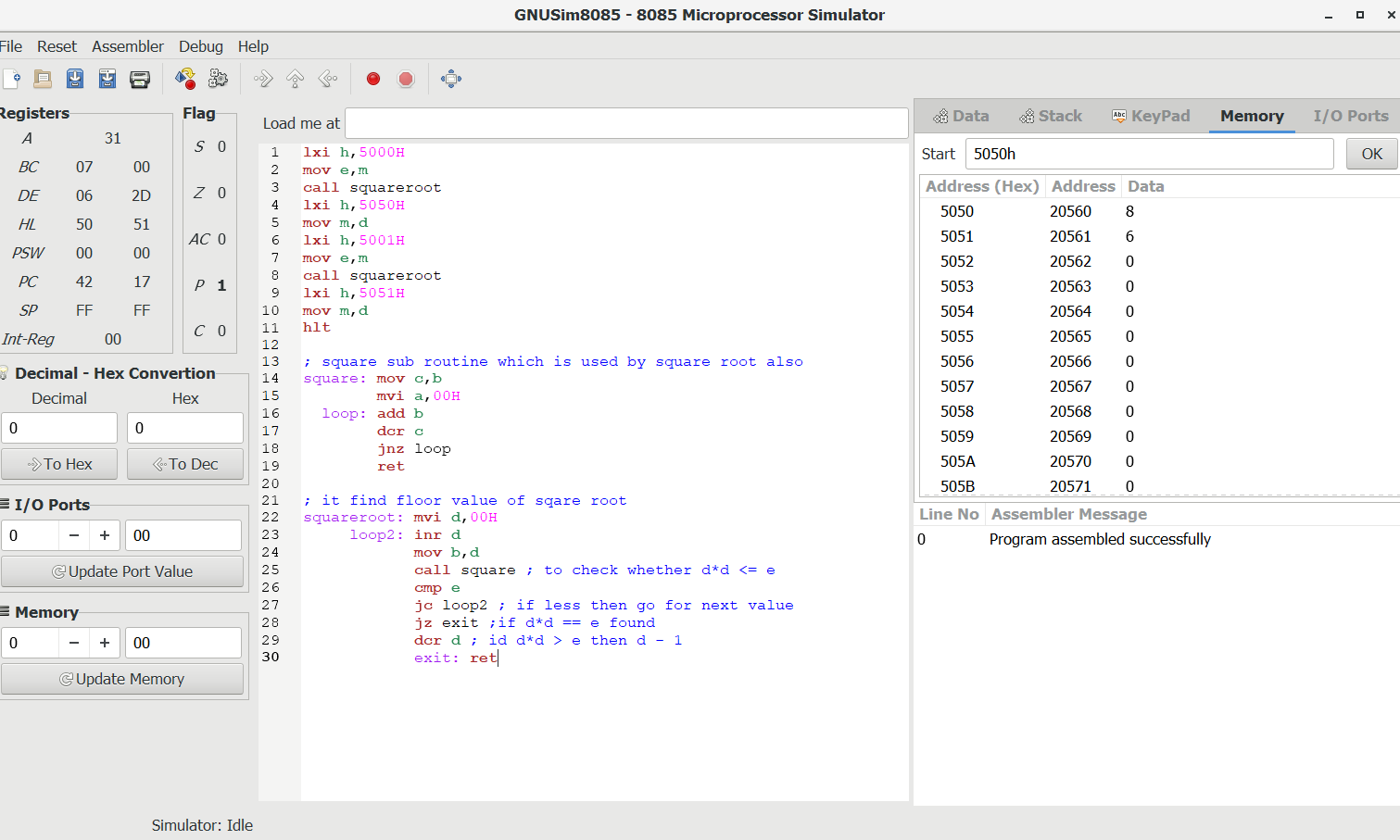
**dcr d ; id d\*d > e then d - 1**

**exit: ret**

**Before : -**

****

**After : -**

****