Johnny Nguyen

November 15, 2022

ECGR 3183 – Computer Organization.

Project #3: More Procedures

Source Code:

```
Main:
  ADDI X0, XZR, #5  // int a = 5
  ADDI X1, XZR, #10 // int b = 10
  ADDI X2, XZR, #8  // int c = 8
  ADDI X3, XZR, #2  // int d = 2
  ADDI X4, XZR, #5  // int e = 5
  ADDI X5, XZR, #10 // int f = 10
  ADDI X6, XZR, #7  // int g = 7
  ADDI X7, XZR, #12 // int h = 12

  // Save stacks for Average.
  SUBI X28, X28, #24
  STUR X22, [X28, #16]
  STUR X20, [X28, #8]
  STUR X21, [X28, #0]
  BL Average // Call procedure, Average.
  B Exit

Average:
  // Sums of parameter registers, 59.
  ADD X21, X0, X1
  ADD X21, X21, X2
  ADD X21, X21, X3
  ADD X21, X21, X4
  ADD X21, X21, X5
```

```
  ADD X21, X21, X6
  ADD X21, X21, X7
  ADDI X20, XZR, #8  // int s = 8
  // Save stacks for Divide.
  SUBI X28, X28, #16
  STUR X30, [X28, #0]
  STUR X19, [X28, #8]
  BL Divide // Call procedure, Divide
  BR X30 // Return to Main.

Divide:
  ADDI X19, XZR, #0 // int r = 0
Loop:
  SUBS XZR, X21, X20  // while(a >= b){}
  B.LE Return
  SUB X21, X21, X20 // a = a - b
  ADDI X19, X19, #1 // r++
  B Loop
Return:
  ADD X22, X19, XZR  // return r
  // Restore stacks from Divide.
  LDUR X19, [X28, #8]
  LDUR X30, [X28, #0]
  ADDI X28, X28, #16
  BR X30 // Return to Average.

Exit:
  ADD X0, XZR, X22 // int i = average(a,b,c,d,e,f,g,h)
  // Restore stacks from Average
  LDUR X21, [X28, #0]
  LDUR X20, [X28, #8]
  LDUR X22, [X28, #16]
```

ADDI X28, X28, #24

ADDI X0, XZR, #0 // Return 0

Simulation:

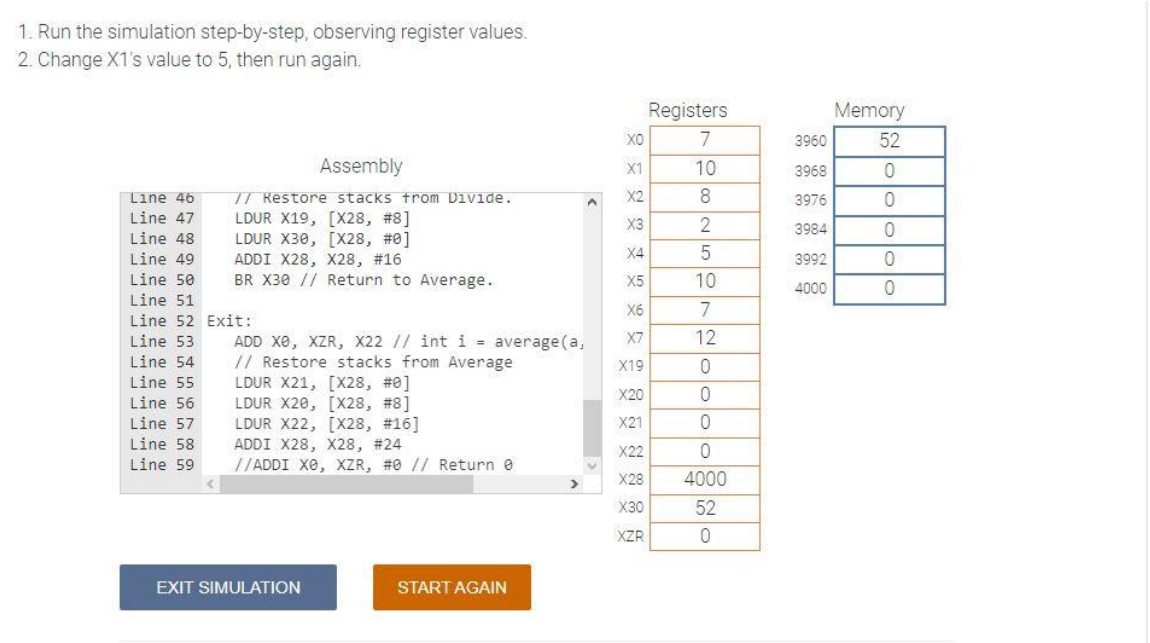

Figure 1: Without return zero, X0 is 7 (the answer)

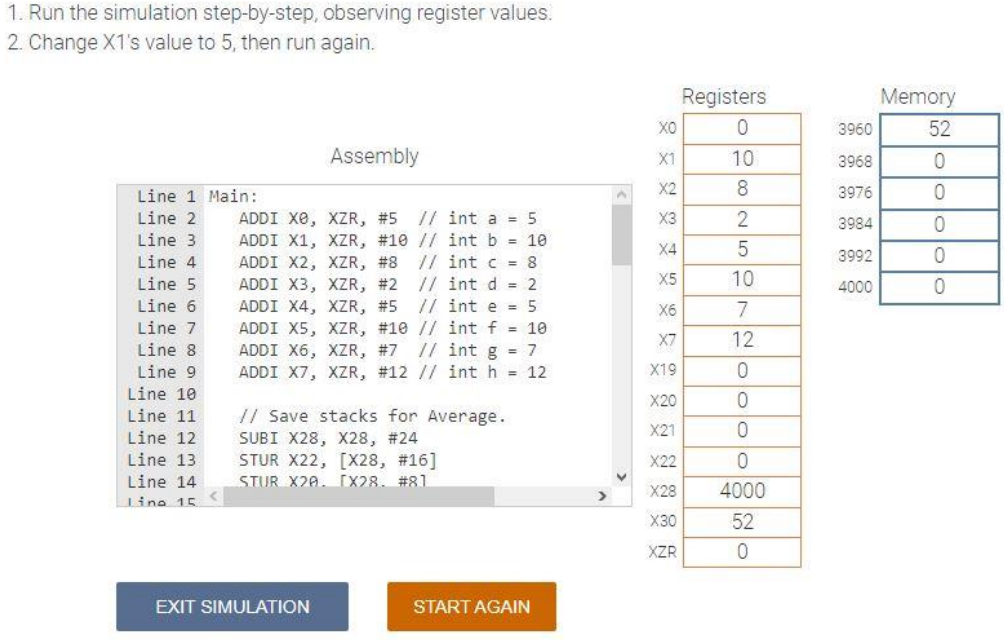

Figure 2: With return zero, X0 is 0. End of Program.