

**Johnny Nguyen**

**October 30, 2022**

**ECGR 3183 – Computer Organization.**

**Project #2: Branching Conditions and Procedures**

Source Code:

Main:

```
ADDI X1, XZR, #5 // int j = 5;
```

```
ADDI X2, XZR, #3 // int k = 3;
```

```
SUBI X28, X28, #16 // Allocating stacks
```

```
STUR X9, [X28, #8]
```

```
STUR X19, [X28, #0]
```

```
BL Multiply
```

```
ADD X20, X0, XZR // int h = Multiply(j,k);
```

```
B End
```

Multiply: // int Multiply(a,b) {};

```
ADDI X9, XZR, #0 // int i = 0
```

```
ADDI X19, XZR, #0 //int r = 0
```

Loop:

```
SUBS XZR, X9, X2 // while (i < b){};
```

```
B.GE Return // Exit the loop, if i == b.
```

```
ADD X19, X19, X1
```

```
ADDI X9, X9, #1
```

```
B Loop
```

Return:

```
ADD X0, X19, XZR // Stores X19 into X0 for return.
```

```
LDUR X19, [X28, #0] // Deallocating stacks
```

```
LDUR X9, [X28, #8]
```

ADDI X28, X28, #16

BR X30 // Return address

End:

ADDI X0, XZR, #0 // return 0;

Simulation:

Assembly

```
Line 1 Main:
Line 2   ADDI X1, XZR, #5 // int j = 5;
Line 3   ADDI X2, XZR, #3 // int k = 3;
Line 4
Line 5   SUBI X28, X28, #16 // Allocating stack
Line 6   STUR X9, [X28, #8]
Line 7   STUR X19, [X28, #0]
Line 8
Line 9   BL Multiply
Line 10  ADD X20, X0, XZR // int h = Multiply(a,b)
Line 11  B End
Line 12
Line 13 Multiply: // int Multiply(a,b) {};
Line 14   ADDI X9, XZR, #0 // int i = 0;
Line 15
```

Registers

X0	0
X1	5
X2	3
X9	0
X19	0
X20	15
X28	4000
X30	24
XZR	0

Memory

3984	0
3992	0
4000	0

EXIT SIMULATION

START AGAIN