Zach Shen (zs255)

Eric Berg (eb645)

ECE3140

Lab Report 1

In the fib function, when n is not 0, or 1, recursion is used to evaluate the nth Fibonacci number. Before the recursive process begins, the current LR memory address must be pushed to the stack, so that it does not get overwritten when fib is called again. Next, R0 which holds the value of n-1 must also be pushed to the stack because the next fib function call will overwrite R0. Because R0 still holds the value of n-1, the fib function is called to evaluate the fib(n-1). The stack will grow as the fib function is called until R0 equals 1, then all the R0 and LR values will be popped again until there are only the original LR and R0 values which were first pushed to the stack. Next, the n-1 value stored at the top of the stack is popped and stored into R1. Next, R0 must be pushed to the stack again to save the value of fib(n-1). Next R0 is set to n-2 using R1. Again, the fib function is called to evaluate the fib(n-2). Now R0 holds the value of fib(n-2) . Next we must POP the fib(n-1) value at the top of the stack into R1. Finally, the two values stored in R0 and R1 are added together and stored in R0 (serving as return value). Finally, the stack pointer is at the LR memory address stored originally and popped into LR. Now that the nth Fibonacci numbers has been calculated the program branches back to main.

Stack (higher address) N=3

|  |  |  |  |
| --- | --- | --- | --- |
| **Label** | **Stack** | **Comment** | **Updates to Stack Cell** |
| Main Activation Record | Main’s stuff |  |  |
| Fib activation record | LR | Fib’s return address to main | POP into LR once fib(n-1) + fib(n-2) has been calculated |
| R0 | Fib’s input parameter, | when pushed value is n-1, after fib(n-1) is evaluated, n-1, is popped into R1, then the RO, which has value fib(n-1) is stored in stack, then when fib (n-2) is evaluated, Pop its value into R1 |
| First Nested fib activation record | LR | First Nested fib’s return address |  |
|  | R0 | First Nested fib’s input parameter |  |
| Second Nested fib activation record | LR | Second Nested fib’s return address |  |
|  | R0 | Second Nested fib’s input parameter |  |
| Third Nested fib activation record | LR | Third Nested fib’s return address |  |
|  | R0 | Third Nested fib’s input parameter |  |
|  | ….. More nested fib activation records | | |
| Final Nested fib activation record (occurs when n = 1) | LR | Final Nested fib’s return address, | Popped when fib(1) is calculated |
|  | R0 | Final Nested fib’s input parameter, | Value of R0 is |

Work Distribution