

ECSE 324
Laboratory No. 2 Report

Ismail Faruk 260663521

Tanbin Chowdhury 260578441

Introduction

This is the report for Lab 2. The lab assignment given was to program some tasks in assembly and implement one of them in C.

Part 1: Subroutines

Part 1.1: The stack (stack.s)

Description

This part was about implementing a PUSH and POP instruction using an alternate way.

Approach Taken

We implemented the STMDB and LDMIA commands.

Challenge Faced

None.

Part 1.2: The subroutine calling convention (subroutine.s)

Description

This part was about converting the program for finding the max of an array into a program that uses subroutine.

Approach Taken

Implemented a stack to store the arguments and reset the registers after the task was performed. Branching using the subroutine calling convention was done from _start. The instructions which compared and stored the maximum value was all within the subroutine.

Challenge Faced

None.

Part 1.3 Fibonacci calculation using recursive subroutine calls (fibonacci.s)

Description

This part was an assembly program which computes the nth Fibonacci number.

Approach Taken

A recursive subroutine was used as instructed. The nth Fibonacci number and the Load Register would always be preserved by storing them in stack. The value of the sub-result would also be stored in stack. This way, the recursion could reuse registers.

Challenge Faced

Finding the right way of storing and changing the value of the sub-result from recursion. It was very tricky to find out.

Improvements made

Multiple stack push and pull commands were present within the recursion to preserve the data.

Part 2: C Programming

Part 2.1: Pure C (max_pureC.c)

Description

This part was about creating a C program and filling in the missing iteration.

Approach Taken

A simple count controlled iteration was used.

Challenge Faced

Variables had to be declared separately. Did not know that.

Part 2.2: Calling an assembly subroutine from C (max_2_subroutine_test.c, max_2_subroutine.s, max_2_task.c)

Description

This part was about creating a C program and calling an assemble subroutine.

Approach Taken

The assembly file to compare two numbers was created. The C program to call that subroutine was also created. Then an updated version was created to find the maximum value in an array by calling the assembly subroutine.

Challenge Faced

None.

Conclusion

The lab requirements were very straight forward. The interesting part was using stack to preserve the initial arguments and calling assembly subroutine using a C program.