XC

CS 200: Computer Organization

Project 10: Maze Generator 2

Shariq M. Jamil

Due: Monday, April 28, 2014

Overview

Purpose

This project required us to write a procedure in MIPS assembly language that recursively generates a maze using functions created in the previous project.

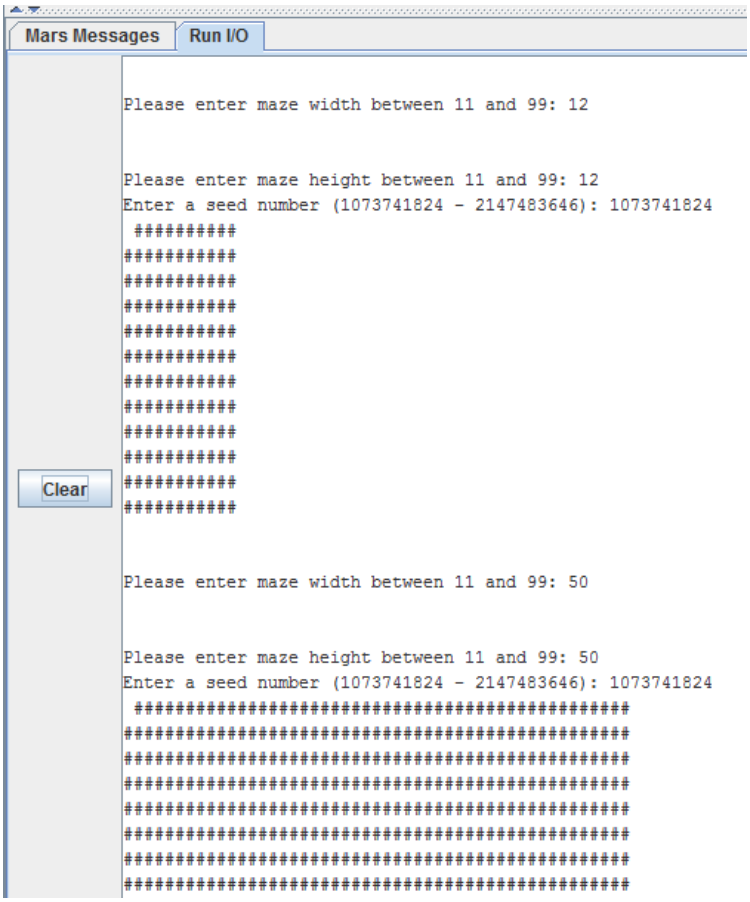
Approach

The project 10 skeleton file was very useful in this project. I followed the project guide and used the skeleton to write out the Visit function. I broke the C-implementation of Visit into smaller parts and converted them to pseudo-code. This made it easier to come up with MIPS implementations for each little part and get them to work together in the Visit. I tested each of the sub-procedures I created to ensure that data was entering and leaving the way I wanted it too. This was also to ensure that the jumps, if-statements and loops were working correctly. I kept getting stuck in never ending loops. I also used functions I created for the Random Number Generator to take in values for a custom height and width for the extra credit portion of the project. This part was my favorite part of the project because I could test the result by looking at the grid and printing out the multiplicative result of height and width.

Solution

Sample Output

Maze with various sizes based on input.



Conclusion

This project was another great exercise in test-driven programming in assembly. The maze would not generate properly if any part of Visit was not functioning properly. I had to come up with tests for each function to troubleshoot and ensure correctness of the procedure. I had a lot of trouble getting the maze to iterate through because my second for loop refused to work. This project was also a good practice in understanding code in a certain language and then translating it into a completely different language. I feel that

this kind of project makes me feel more confident in my ability to grasp new languages and technologies quickly and be adept at them.

References

MIPS Architecture and Assembly Language

http://logos.cs.uic.edu/366/notes/mips%20quick%20tutorial.htm

Understanding the Stack

http://www.cs.umd.edu/class/sum2003/cmsc311/Notes/Mips/stack.html