

CSCE 2610-001

Michael Mohler

Ka Son Chan

March 30, 2012

### Project 1 WriteUp

I always separate the programming assignment into functions/section when I work on any programming assignment. In this project, I work on individual functions before combining them together to one big project.

First challenge is that I am super lazy to start working on it and I need to read many specific examples online and from the textbook. That's the way I can learn a new programming language fast and solve my assignment. Also, I'm quite familiar with C++/C. I can learn the syntax and semantics of MIPS from the specific examples from translating from C++/C to MIPS online.

At the beginning, when I try to make the command loop works. I don't know how to make a loop - after reading examples, I learn it by jumping around several labels. After making the Valid user command loop works, I move forward to write small program - getting character/number/string from user and displaying them. At that stage, I struggle to printing the strings - need to use `li $a1, <how many space>`. Again, I surf around the web to find the way to solve it.

After that, I try to make the getting and printing char/number/string by jumping around different labels (call different functions). For the function part, it went pretty smooth - don't have problems. Then I try to write a linked list. I read the `dynamics.asm` that you posted on blackboard and I quite understand it. But don't know how to make the root and link the linked list to the root. I took a few hours to figure it out again by surfing online. Next, I try to get some char/number/string from user store in linked list and displaying them. This stage went pretty smooth. After that, I try the same thing using different function calls (labels). It was quite a mess. I mess up the `jal` and `jr $ra` commands. `jr $ra` means back, and `jal` means call with link!

The next big challenge is the recursion part. I try to make my little program of recursion. It was a difficult because I can't really find good example online to get me through. I spent many days to figure it out. I always mess up `$a0` and `$v0`. The `$a0` stores the data, and `$v0` stores address. Also, I didn't really understand when to return back to the caller and when not. After

working on this project, I'm pretty understand it now.

When I am assembling my little programs to one big project. I mess up when I forgot to change back the memory allocation to the project structure. It is hard to find when debugging. Many times I messed up when I assemble the programs is that I forgot the change the address from the little programs to suit the big project. Working through this project, I learn a lot writing MIPS assembly language. Especially how to handle register and addresses; when to return back to the caller in recursion, and how to create pointers and addresses.

To sum up, I complete this project by writing some small-function programs, browsing internet and understanding some unknown information online. The procedure of working through this project is first creating a main function, main looping function, add functions, printSmall and printLarge functions, preorder and inorder recursive traversal functions, and finally the find functions. The main problems that I encountered is I didn't understand the MIPS language - save data to wrong registers. I learnt a lot working on this project - let user input data, output data, handling registers and addresses; when to return back to the caller in recursion, and how to create pointers with addresses.