COL216 A-3 Write-up

Sayam Sethi 2019CS10399
Mallika Prabhakar 2019CS50440

March 2021

Problem Interpretation

Assumptions

Following are the assumptions which have been considered or given while attempting this assignment:

- Input file is a text file.
- Instruction format follows the MIPS convention. The only valid instructions are- add, sub, mul, beq, bne, slt, j, lw, sw, addi.
- Memory is 2^{20} Bytes and we have 32 registers.
- Only the values printed for registers are in hexadecimal form.
- Each instruction occupies 4 bytes and is executed in one clock cycle.

Basic Idea

We have to implement a C++ program which takes a text file as input which contains MIPS Assembly instructions. We read it line by line and tokenize each line appropriately. We then figure out the command and operate accordingly while updating the registers and incriminating clock cycles along with how many times certain operation was called. This information is finally printed. Errors are handled accordingly.

Code Explanation

Following is the list of procedures employed along with their functioning:

- **MIPS_Architecture** - constructor
- **add** - addition operation
- **addi** - addi operation

- **sub** - subtraction operation
- **mul** - multiplication operation
- **op** - perform the operation
- **beq** - perform the beq operation
- **bne** - perform the bne operation
- **bOP** - implements beq and bne by taking the comparator
- **slt** - performs slt operation
- **j** - perform the jump operation
- **lw** - perform load word operation
- **sw** - perform store word operation
- **locateAddress** - locates the address
- **checkLabel** - checks if label is valid
- **checkRegister** - checks if the register is a valid one
- **checkRegisters** - checks the validity of all the registers
- **handleExit** - Handles possible errors according to output code
- **parseCommand** - parses one line to figure out the command
- **constructCommands** - parses the entire input file
- **executeCommands** - runs the commands
- **printRegisters** - prints clock cycle and values of registers
- **main** - takes file as an input and executes the MIPS commands using all the functions mentioned above

Testing

We have extensively tested our code broadly on the following test cases
case 1: empty file input
case 2: random input
case 3: for loop
case 4: while loop
case 5: erroneous input