CS350 CS402 ILLINOIS INSTITUTE OF TECHNOLOGY Summer, 2018

Project Description

Instructor: Dr. Michael Choi
TA: Chaoqi Ma

1. Project Overview

Create a MIPS numerical calculator. Refer to textbook chapter 2 and 3 for MIPS calculation and implementation.

2. Project Requirements

This project must be done under following environments:

Programming tool: **SPIM**

Virus Free Project

Under IIT department of Computer Science Policy, any project submission with virus should be excluded from the evaluation. It means, TA will not continue to evaluate your project material. The final grade of project will be 0. To avoid any potential debate, everyone should follow the department rule. Thus before you submit your project, you have to check your package with the latest antivirus software at multiple computers.

3. Description of the Project

User Interface: This project does not mandate a specific user interface. Thus you can freely design your own user interface. *Below screen shots are just samples and you don't need to follow the same way.*

Main Body: When your project starts, there are options for arithmetic operation such as Addition, Subtraction, Multiplication, and Division and operand numbers can be entered. User interface can be designed as your preferred way such as: (below shows just some examples)

- human way such as "X + Y"
- Like MIPS assembly order: (e.g.) when Menu Subtract chosen, X Y
  Or
  First operand: X
  Second operand: Y
- Or Menu driven way:

- Example (no need to build the same menu interface – **Create your own menu or calculator screen**)

```
MIPS Calculator
============
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Conversions

0. Exit
```

Then, get inputs of operands such as X and Y.

Implementation requirements

Each arithmetic operation must be called out by a corresponding procedure from the main body. So MIPS assembler coding should include procedure calling parts and a remark or a comment must be included.

Main project requirements:

1. Get user selection menu which operation to perform basic arithmetic operation: Addition, Subtraction, Multiplication, or Division (Any interface format is fine but try human way ‘x + y’ or similar) and extras.

2. Calculator does not end unless the user selects Exit or required selection. Thus the program needs to go back to the beginning unless the user selects EXIT.

3. Reliable and robustness: if an incorrect input is entered such as “1 + &”, then a proper error message should be returned and keep continue to get next correct input instead of crash the calculator project.

4. Unless menu 0 is selected, the program does not end but keep working another operation by selecting menu.

Menu 5: Conversions

This selection provides extra features as you can find your scientific/engineering calculator. Those required features are:

```
MIPS Conversions
================
1. Decimal to Binary
2. Decimal to Hexa
3. Binary to Decimal
4. Binary to Hexa
5. Hexa to Decimal
6. Hexa to Binary
7. To Main Menu
0. Exit
```

1. You understand all other features.
2. Conversion menu 1 to 6 returns conversion between the number formats.
   e.g) decimal 5 returns to binary value of 101 by menu 1.
3. Selection 7 returns to the main menu: basic arithmetic operations
4. Selection 0 exits from the calculator (same from the main menu)

<u>What to submit?</u>

To the Course Blackboard project tab:

1. Source MIPS assembly codes (must be executable by SPIM)
   a. All source code must include detail comment of what the code does
   b. Detail comment is a good habit of documentation that professionals in the SW industry are required
   c. Good documentation means quality software in the industry

2. User’s manual document: How to use/run your project (include Program analyzing document) – because everyone’s interface is different so everyone needs to have a user documentation how to run your project
3. Final Design diagram document (or flow charts)
4. Self evaluation doc – describe what features are not yet implemented. If your project has all the features, no need to include it. But if your project does not include any feature, describe the item(s) in this document so your TA does not test it (them). TA will apply default penalty of each unimplemented feature/requirement. However, if your TA finds any missing feature but not documented in your self evaluation, more penalty may be given.

All above must be in one zipped file (filename.zip).

** Only use **zip format**. **No other format such as RAR.**

Final Project Due Date: ***June 24, 2018 Sunday midnight*** or earlier

Submit to the class BB (blackboard)
Late submission penalty: 10% every hour
e.g.) if you submit it at 5:01 AM, your max score will be 40%
of the project

Demonstrate your project to your TA on either **6/25 or 6/26**
TA will announce your demonstration schedule

- No demo after final exam is permitted
- **Final exam means the end of a summer semester**