Timestamp: 2020/09/03 22:26

CSC2002S: Architecture 1

MIPS Assembler Programming

Introduction

This assignment is about assembly language programming using the MIPS simulator called SPIM Open source versions of SPIM, e.g QtSpim, are available for Windows, Linux, and Mac, and should be able to run on very basic computers. To learn the basics of MIPS assembly language programming, read A.9 and A.10 in the appendix from Patterson+Hennessy and a MIPS tutorial by Daniel J. Ellard (both available in > Resources > Architecture > Additional resources) , and watch the tutorial on https://bit.ly/2ELYa43.

Question 1 [30 marks]

Write a program (`question1.asm`) to enter a series of 5 arbitrary strings and then print them out. Store the strings in memory.

Sample IO:

```
Enter a series of 5 formulae:
1
2
3
=1
=2
```

The values are:

```
1
2
3
=1
=2
```

Question 2 [30 marks]

Write a program (`question2.asm`) to convert a string into an integer.

Assume that your input string is prefixed with an arbitrary non-numeric character that must be ignored and that all subsequent characters are numerals. Add 5 to the input value and print out the sum.

Sample IO:

```
Enter a string:
X123
The value +5 is:
128
```

Question 3 [40 marks]

Using the previous 2 questions, write a program (question3.asm) to compute a simple 1-dimensional spreadsheet containing integers and integer formulae. First, input the spreadsheet source values

CONTINUED

from the console, then perform calculations as outlined below and finally output the computed spreadsheet.

A cell may contain either an integer value or a formula that is a reference to another cell. Cells are numbered 0-4 from top to bottom and a reference is indicated by a "=" prefix followed by the cell number (e.g., =2). A reference can only be made to a previous cell, and the cell referred to may itself contain a formula.

Sample IO:

```
Enter a series of 5 formulae:
1
2
3
=1
=2
The values are:
1
2
3
2
3
```

Submission

Submit ALL asm files in a single ZIP file to the Automatic Marker. Ensure that your code includes sufficient commenting.