CSCI 250, Programming Assignment 1
Spring 2011
20 Points Total

1 Background

The *Peasant Method* is a method for multiplying two integers (multiplier × multiplicand). For each 1 bit in the multiplier, we produce a new integer which is the multiplicand shifted to the left p places, where p is the position of the 1 bit in the multiplier. We then sum each shifted integer.

The reason the method is called *Peasant Method* is because it was used by peasants who were generally unschooled and thus unfamiliar or incapable of memorizing multiplication tables required by long multiplication. The method was also in use in ancient Egypt:

http://en.wikipedia.org/wiki/Multiplication_algorithm#Peasant_or_binary_multiplication

I multiplied two integers in class using this method. This example is also available on the wikipedia page.

2 Objectives

1. To test your knowledge of the MIPS R2000/R3000 assembly language instruction set as well as your ability to appropriately combine MIPS assembly instructions towards a solution of a simple (yet interesting) problem
2. To test your understanding of (and ability to properly execute) the requisite steps for performing a function call (including proper manipulation of the runtime stack)
3. To test your ability to abide by proper register use conventions as described in class (and in the book)
4. To gain experience with SPIM
5. To test your ability to construct a correct loop in MIPS assembly
6. To gain experience with syscalls (for input and output)
7. To gain experience with debugging (stepping through) a simple SPIM program (as required)
8. To test your ability to detect overflow

3 Description

Your task for this programming assignment is to implement the *Peasant Method* in MIPS assembly. Your code must:

- be loadable and runnable in SPIM (use appropriate editor as described in class)
- be formatted similar to the 'hello.s' and 'length.s' examples given in class. Specifically, you must have (in order) the following sections in your assembly program:
  - name of the program
  - 1-3 sentence description of what the program does
  - list of each register used followed by a one-line description of what value this register holds (NOTE: For simple programs like this one there is no need to reuse or overload register names.)
  - the "data" segment (use descriptive names for variables declared here)

  - the "text" segment (here is where you list your code)
  - comment each executable statement appropriately

- the multiply code must be implemented as a **function** which receives two parameters, multiplier and multiplicand, and returns the result of the multiplication (you should document your function in a similar fashion that you commented your 'main')
- prompt for and input the multiplier and multiplicand in the console window
- output the result in the console window
- output whether overflow occurred or not

4 Due Date and Submission

You are to submit your assignment on stono with the following command:

```
$ submit csci250 prog1 filename.s
```

NOTE: The name "filename.s" should be changed to the actual name of your program.
The assignment is due February 17th for the TR class and February 21st for the MW class.