# Week 5 Reinforcement

You should try the exercises and make sure that you have your solutions available during the lectorial in week 6. You may want to work on these in small study groups. These exercises have features that are also needed for the coursework.

As well as working on these exercises, you should also spend some of your individual study time working on your coursework. Remember that it is an individual coursework, but you may ask the tutors questions.

### Part 1 MIPS Assembly language subroutines

Download the program **week5reinfpart1.asm** from Moodle. Assemble and run the program. It asks the user for two numbers (integers) and gives the remainder when dividing each by the other. Test the program with various inputs.

Read the program. Notice that it uses a syscall to read in the numbers from the user which wasn’t covered in the lab session. You might want to read the syscall help and/or look at the concepts materials on using syscalls for input.

You should notice that the program has a lot of repeated code. Your task is to use subroutines to reduce the amount of repetition. This is a similar task to part of the coursework.

### Part 2 MIPS Assembly language branches

1. Download the program **week5reinfpart2.asm** from Moodle.

This program asks the user to input two numbers and it reports which is the higher of the two numbers.

Read and test the program. Make sure that you have tried the first number being larger and the second number being larger and the two being the same.

The program uses branch instructions, there are a lot of varients of branch instructions in the MIPS basic instructions and extended (pseudo) instructions. We are only looking at a few of those variations on this unit. You might want to look at the help for MIPS instructions and/or the concepts materials on Moodle.

* 1. When you understand the program, amend it to use subroutine(s) where appropriate.
  2. The program uses the instruction “b” to branch unconditionally, try to explain why that is needed. What difference does it make if that unconditional branch is changed to a jump?
  3. Compare the program to a version that is written for Little Man Computer. We looked at examples of this in the week 4 labs.

1. Look at the countdown loop program that is given in the concepts slides on branches. That program is very limited as it uses register $a0 for the countdown and just prints the countdown on one line with no spaces between.

Using that program and other example programs, create a countdown program that asks the user for an input number and then counts down from that number to 1, leaving a space (or a tab) between each digit.

You will need to store the input number in a register. You should use subroutines if appropriate.

Test your program.

Please make sure that you are familiar with the coursework and take any general questions to the final lectorial or lab session.