

Lab 02

Alex Griffiths
18001525

Preparation

Question 2

The stored-program concept means that data and instructions are stored in memory. And as data is required, it is loaded from memory. This data is then used by the program, and then any results from operations performed on this data can then be stored back into memory, which can be used later. This is different from storing all data in the registers. Data needs to be loaded from memory, into the registers, or written from a register into memory.

Question 3

All three (calculated values, ASCII characters and instructions) can be found in the memory of MIPS computers. Calculated values and ASCII characters are stored in their byte form. Instructions are stored as machine code that can be loaded by the processor.

Workshop Tasks

Question 2

a)

```
s1 = 0 + 12
s2 = 0 + -2
s3 = 0 + 13
s4 = 0 + 3

t0 = s1 + s2 # t0 = 12 + -2 = 10
t1 = s3 - s4 # t1 = 13 - 3 = 10
s0 = t0 - t1 # s0 = 10 - 10 = 0
```

b) The program outputs the following:

```
26850099200.00000000
... Thank you :)
```

c) No the results are not identical. The expected output should be:

```
The value of f is: 0
Have a nice day :)

... Thank you :)
```

Instead there is a very large number that is output. I believe this is happening because the system call code that is loaded on line 23: `li $v0, 1` is the system call code to print out an integer, but the argument passed to the system call instruction is an ASCII string, not an integer. It then prints out an integer value for

this string, and continues to execute the instructions until it tries again to print out the data pointed to by the extra label. This time however, the syscall code that was loaded was the print_float code. After printing out the float value for this data, it continues executing the rest of the instructions.

Question 4

```
$s7 -> used for saving the return address
$s1 -> used for storing the value 12
$s2 -> used for storing the value -3
$s3 -> used for storing the value 13
$s4 -> used for storing the value 3
$t0 -> used for storing the result of the addition of the values in $s1 and $s2. This
stores the value 10.
$t1 -> used for storing the result of the subtraction of the values in $s3 and $s4.
This stores the value 10.
$s0 -> used for storing the result of the subtraction of the values in $t0 and $t1.
This stores the value 0.
$v0 -> This register is used 3 times to store the syscall code.
$a0 -> This register is used 4 times to store the arguments for syscalls. Three times
it stores the address of a label, and one time it stores an integer that is in $s0
$ra -> used for restoring the return address which was stored in $s7.
$0 -> This register is used when that value 0 is required for an operation.
```