Chris Carr - 17105584

manchester metropolitian university

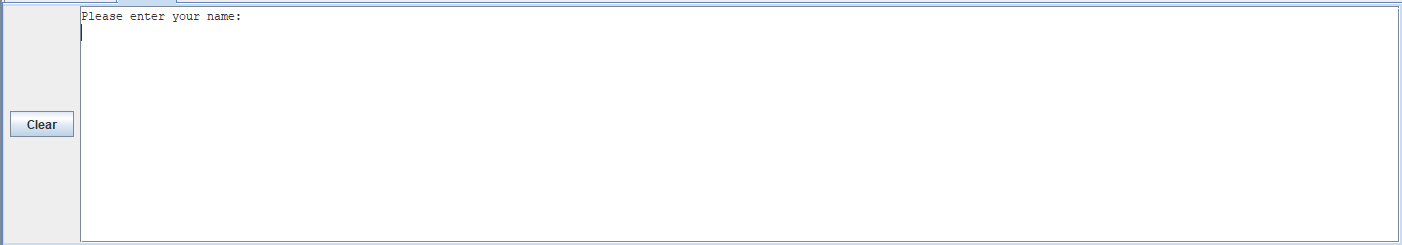
1CWK50 – Task a

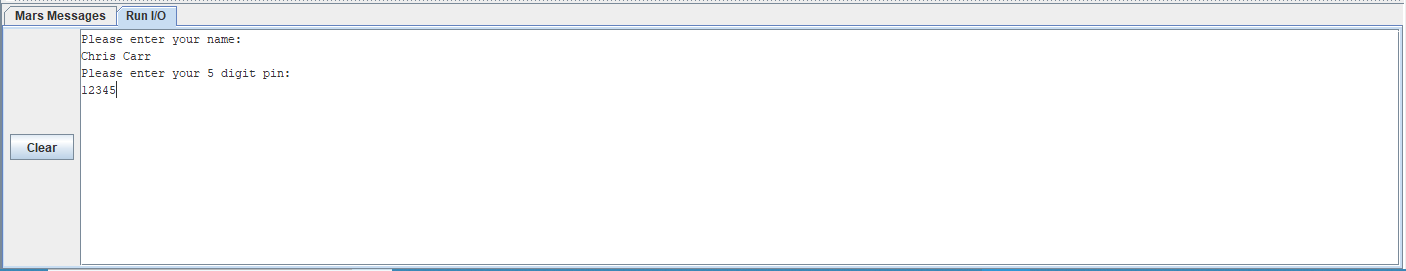
**Task A**

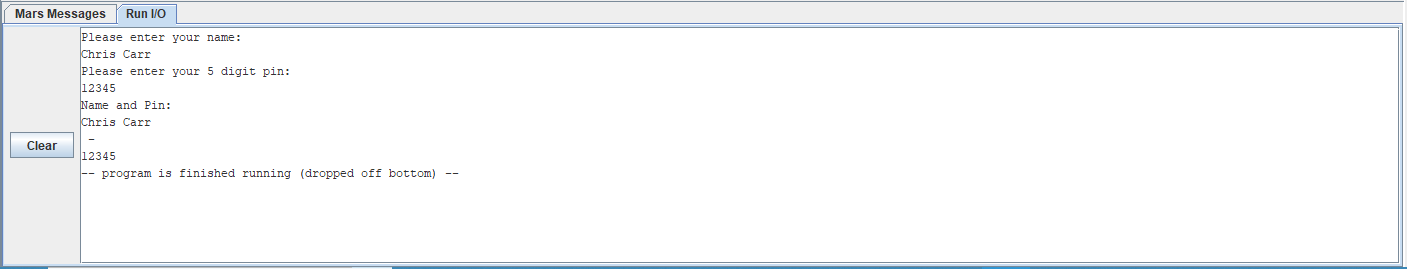
Task A was a good way of reinforcing the information learned through university lectures/labs. It highlighted areas where my basic knowledge was lacking, and I had to resolve these issues with self-learning, rather than asking a tutor for help. It was good to begin the assignment with a basic task. One thing that I noticed throughout this task was that I needed to keep checking the reference sheets (<http://courses.missouristate.edu/kenvollmar/mars/help/syscallhelp.html>), rather than knowing which syscall commands to use off by heart (which the tutors do!). I identified 3 problems with the program:

* The entered string is not validated. For example: names shouldn’t contain numbers or specific special characters (i.e @)
* When the name String is read from the input buffer, a newline is added to the end (\n). This stopped me from displaying the output on a single line, as first intended.
* No validation that the integer is 5 digits long. This could be easily fixed with 2 if statements (input\_pin >= 9999 && input\_pin <= 99999)

*Screenshots*







*Code*

**.data**

**prompt\_name:** **.asciiz** "Please enter your name: \n"

**prompt\_pin:** **.asciiz** "Please enter your 5 digit pin: \n"

**output:** **.asciiz** "Name and Pin: \n"

**separator:** **.asciiz** " - \n"

**.text**

#Display prompt name

**la** $a0, **prompt\_name**

**li** $v0, 4

**syscall**

#Store name input

**li** $v0, 8

**add** $a1, $a1, $a0

##Probably could do some validation here (no numbers) ####

**syscall**

#Display prompt pin

**la** $a0, **prompt\_pin**

**li** $v0, 4

**syscall**

#Store prompt pin

**li** $v0, 5

**syscall**

**add** $a2, $a2, $v0

#Output output

**la** $a0, **output**

**li** $v0, 4

**syscall**

#Output stored name

**la** $a0, ($a1)

**li** $v0, 4

**syscall**

#Output Seperator ("-")

**la** $a0, **separator**

**li** $v0, 4

**syscall**

#Output stored pin

**la** $a0, ($a2)

**li** $v0, 1

**syscall**

#end

Github commit: <https://github.com/jynxmagic/MIPS-Assembly-Language/commit/808a4233c595427e4910d63ab2a691f9e72cf67d>