

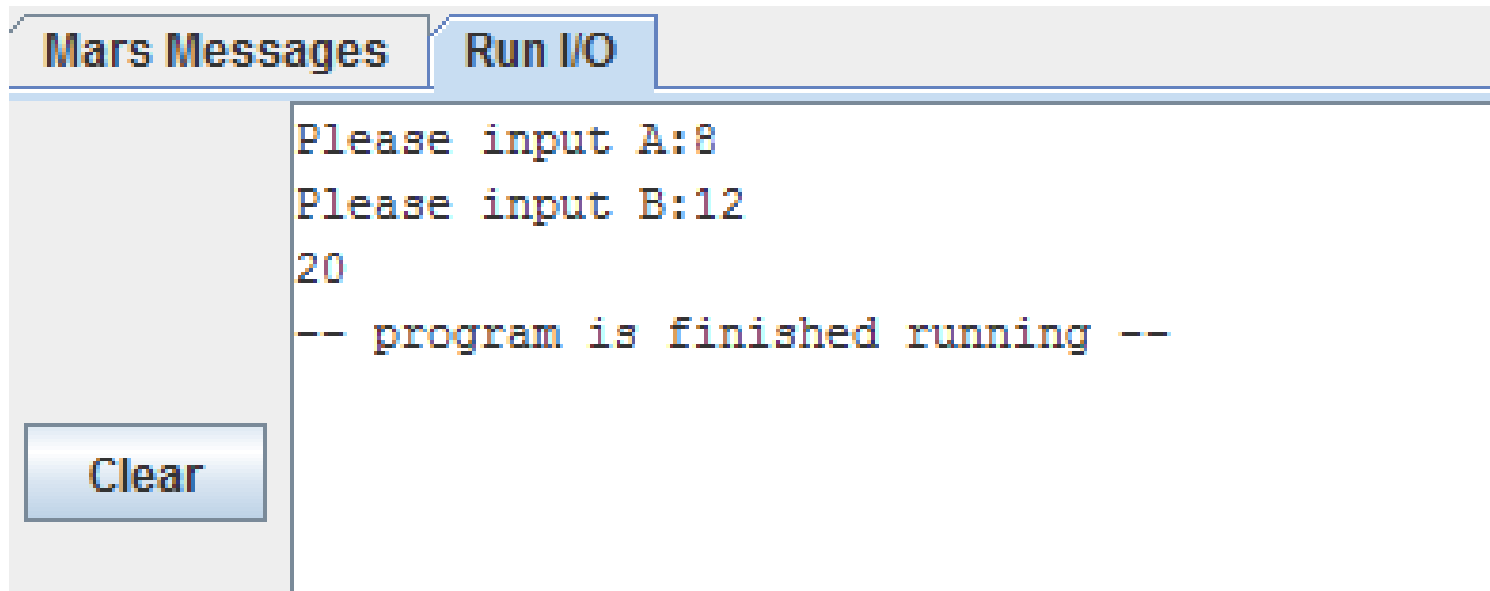
# **COMP2611: Computer Organization**

## **MIPS procedures**

- ❑ You will learn the following in this lab:
  - ❑ how to use MIPS procedures in a program.

- ❑ Try the following example programs in order:
  - ❑ [functionCall1.s](#) (returning through the address in \$ra).
  - ❑ [functionCall2.s](#) (can't return because \$ra has been overwritten).
  - ❑ [functionCall3.s](#) (preserving the registers).
  - ❑ [functionCall4.s](#) (passing more than 4 function arguments).
  - ❑ [functionCall5.s](#) (preserving the function arguments).

- ❑ Please download a file and open it in Mars  
<http://course.cse.ust.hk/comp2611/lab/lab6/addTwoNumber.s>
- ❑ This is a modified version of lab 4 exercise to add two numbers
- ❑ You must call a procedure addNum to perform the addition
- ❑ You may need to preserve the value of \$a0 as it will be overwritten



The screenshot shows the 'Mars Messages' window with the 'Run I/O' tab selected. The output text is as follows:

```
Please input A:8
Please input B:12
20
-- program is finished running --
```

A 'Clear' button is visible in the bottom left corner of the window.

- ❑ You have learnt:
  - ❑ how to use MIPS procedures in a program.