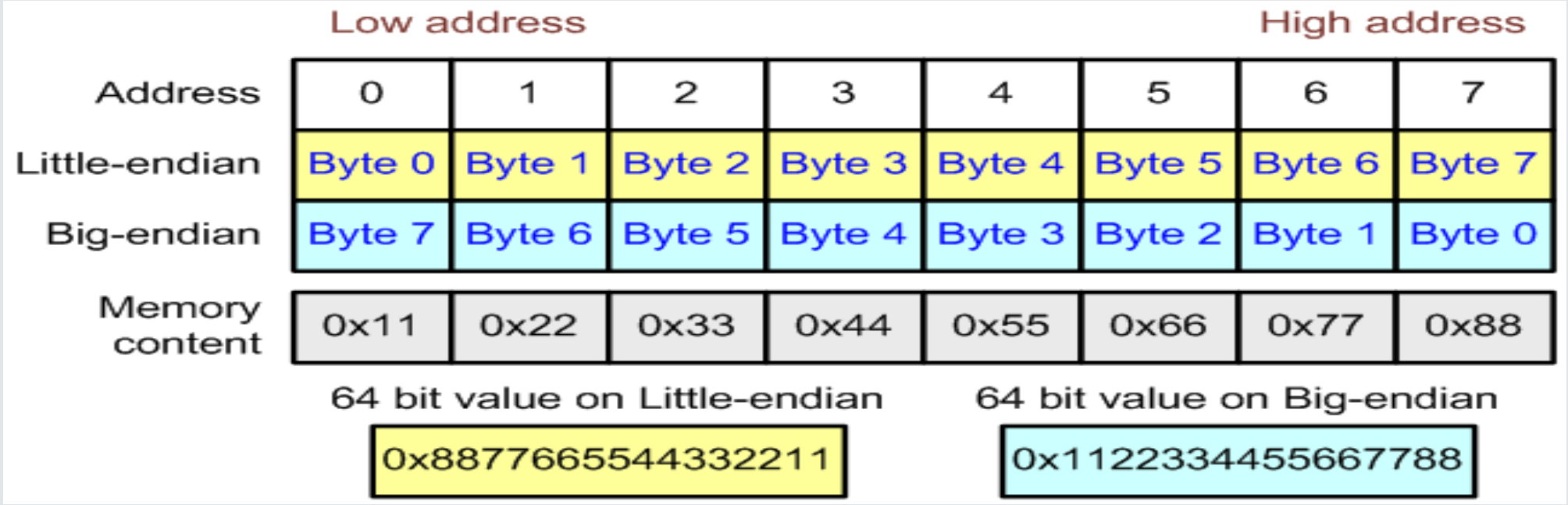
Tutorial- Introduction to MIPS

What is QtSpim?

- Spim is a self-contained simulator that runs MIPS32 programs. (MIPS-Microprocessor without Interlocked Pipeline Stages)
- QTSpim is a new user interface for Spim built on the Qt UI framework.
- Download QTSpim from:- https://sourceforge.net/projects/spimsimulator/files/latest/download

Memory

- Memory is Byte Addressable
- MIPS as simulated in QTSpim is little-endian



QTSpim Program Formats

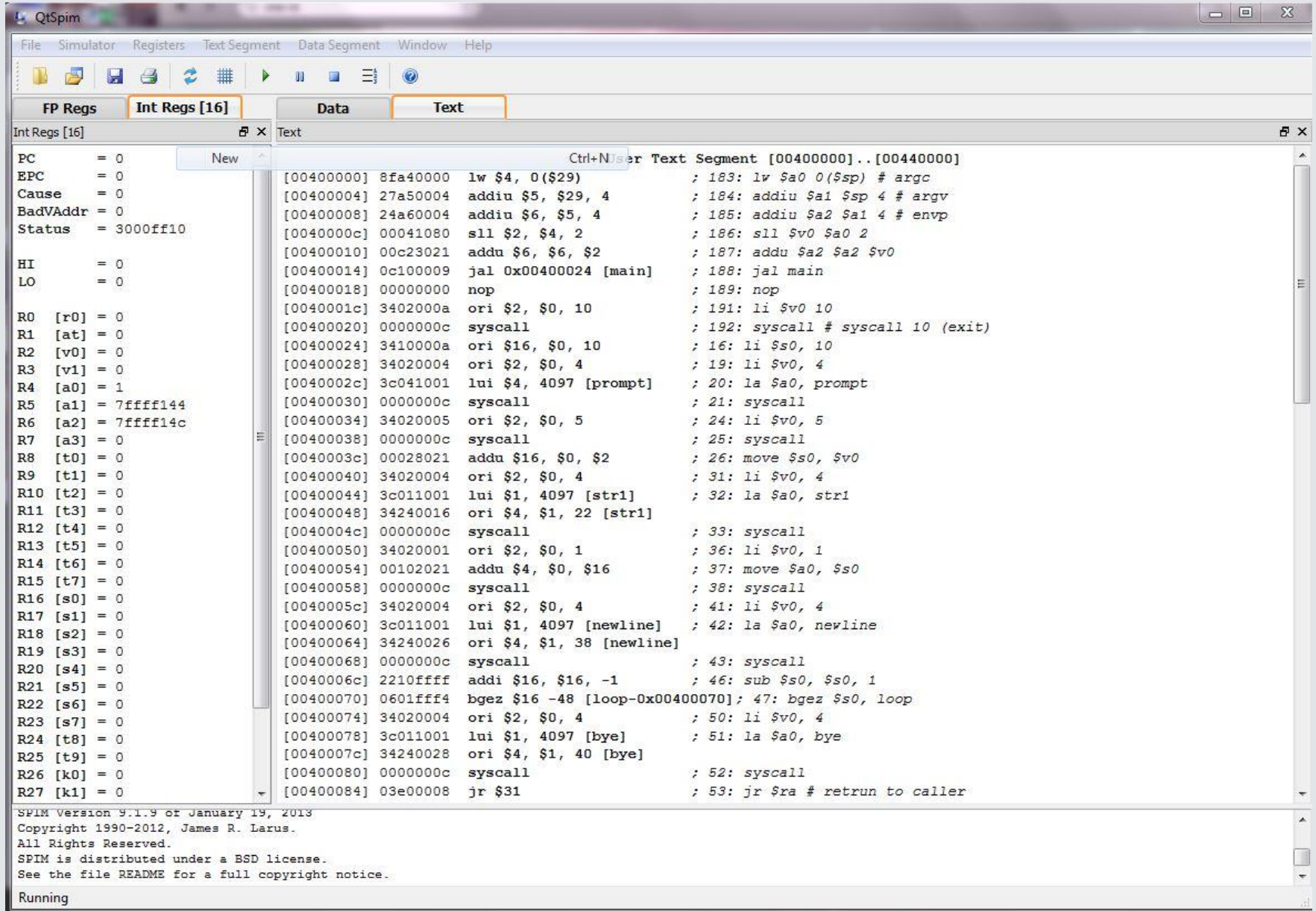
- For comment line we used ‘#’ symbol.
- Assembly directive informs the assembler for data declarations and also where is the text and code. It begins with a ”.’
- The general format for data declaration is :-

<variableName>: .<dataType> <initialValue>

QTSpim: Loading and Executing Programs

- Start QTSpim

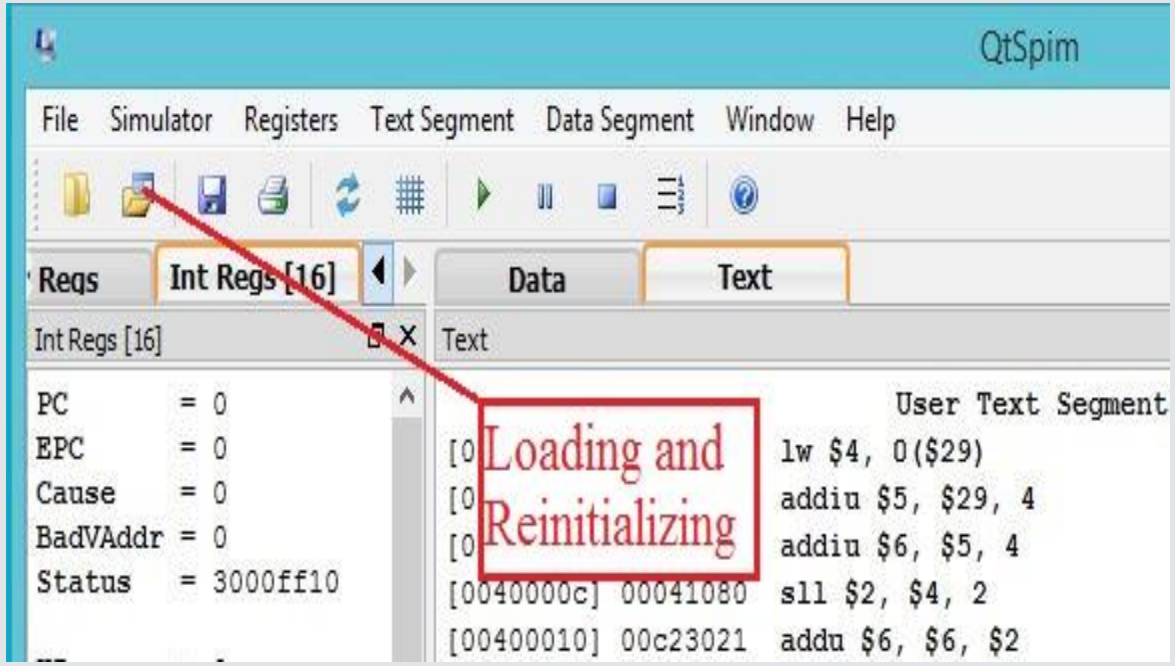
- The initial QtSpim screen will appear as shown.



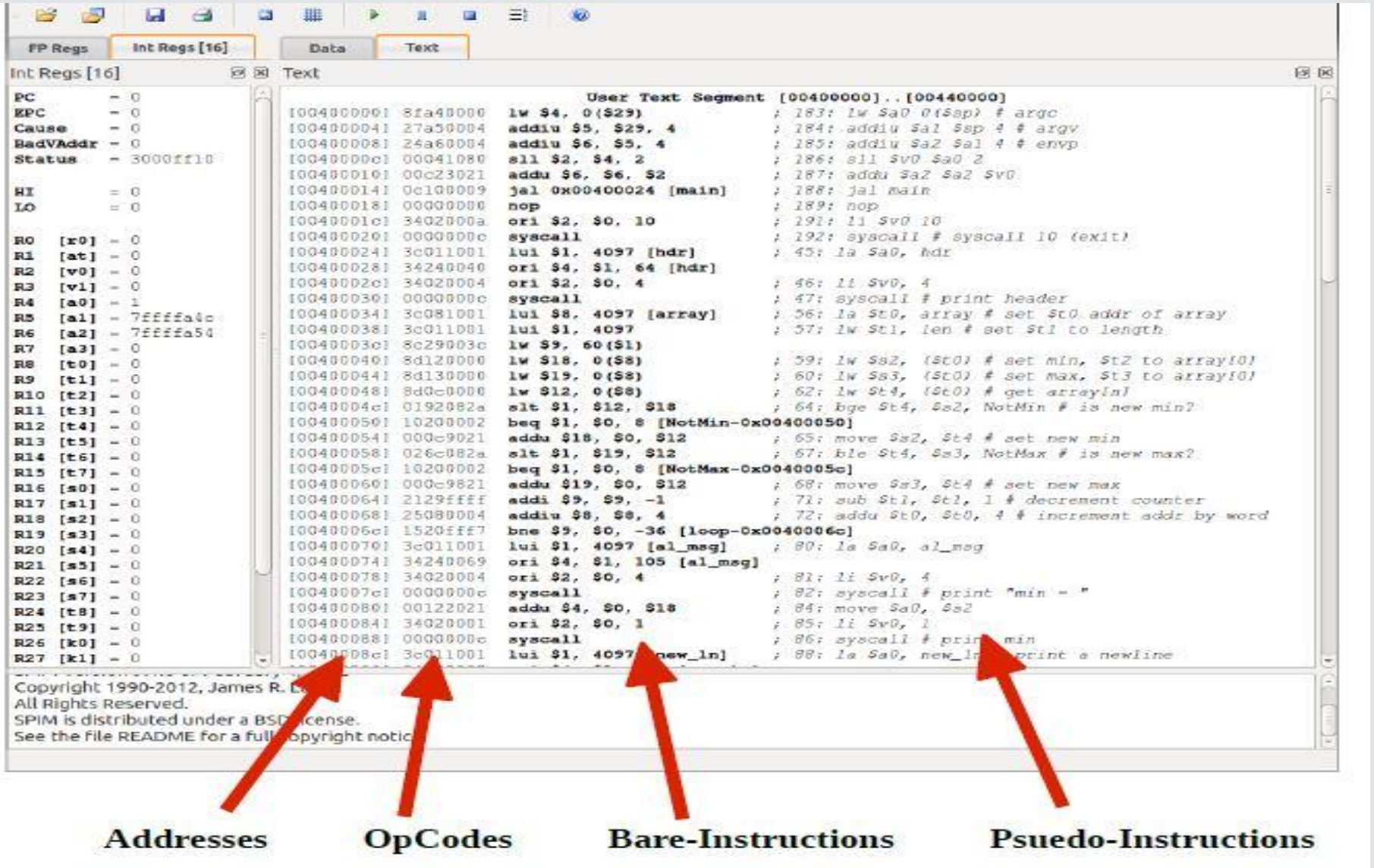
Load & Reinitialize

- To load and reinitialize click on the button highlighted.

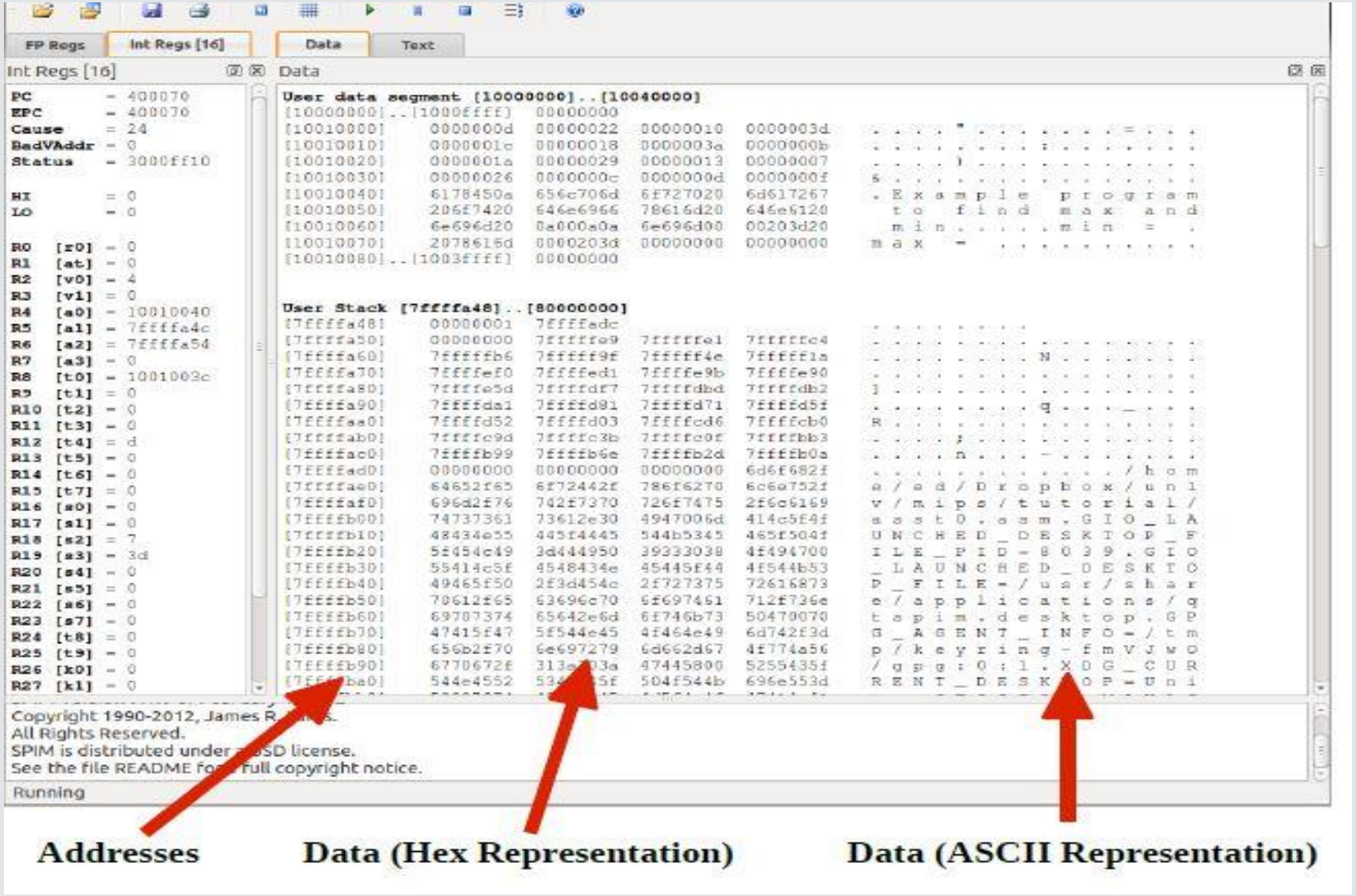
- Once selected, a standard open file dialog box will be displayed. Select the program to be executed



Text Segment

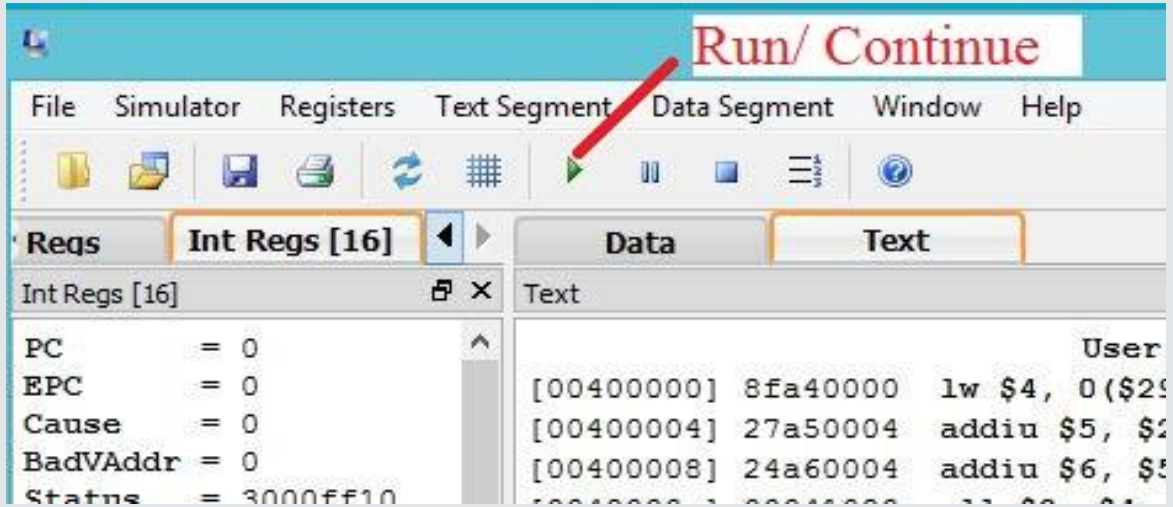


Data Segment



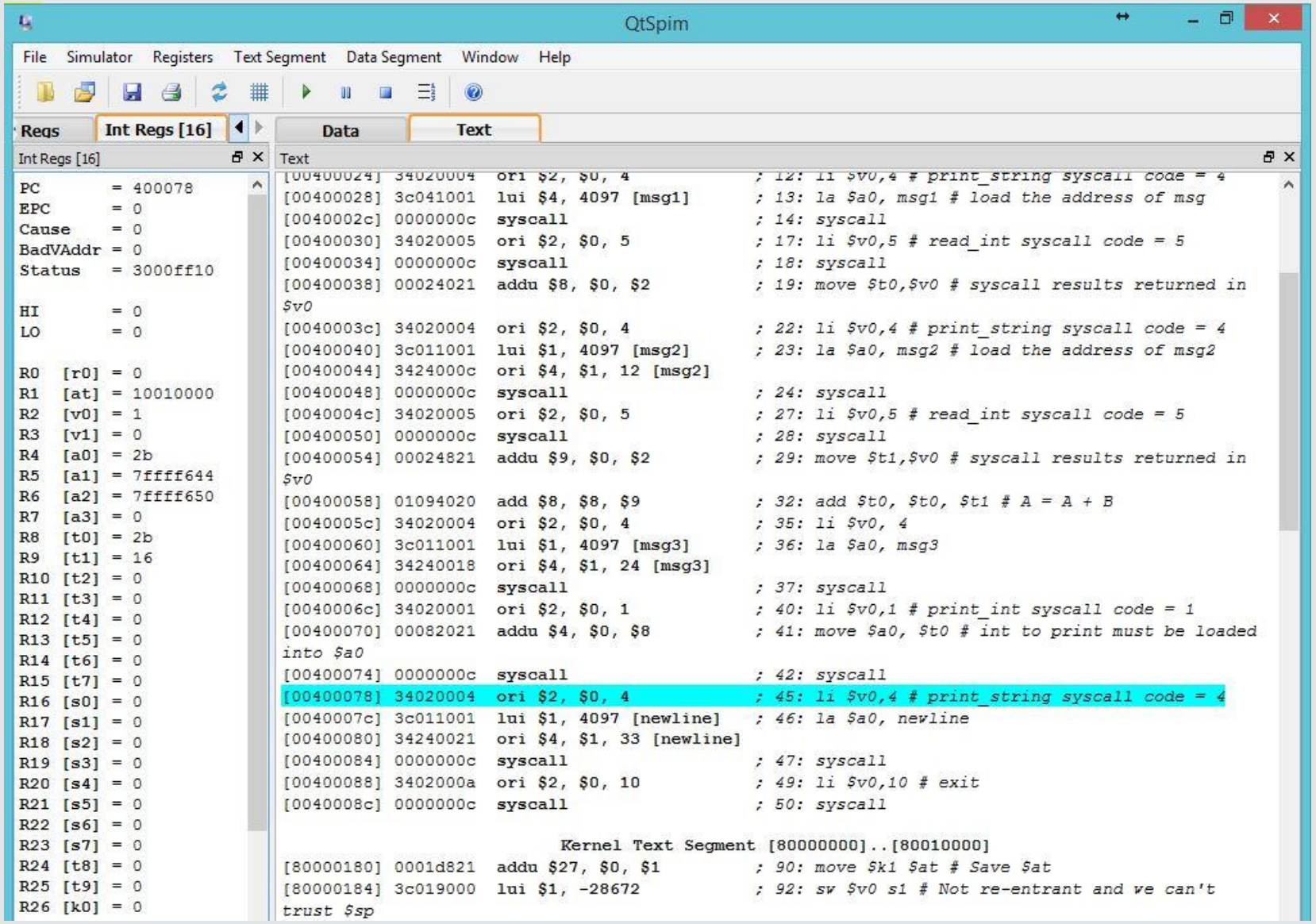
Execution

- To run the program click on the button highlighted

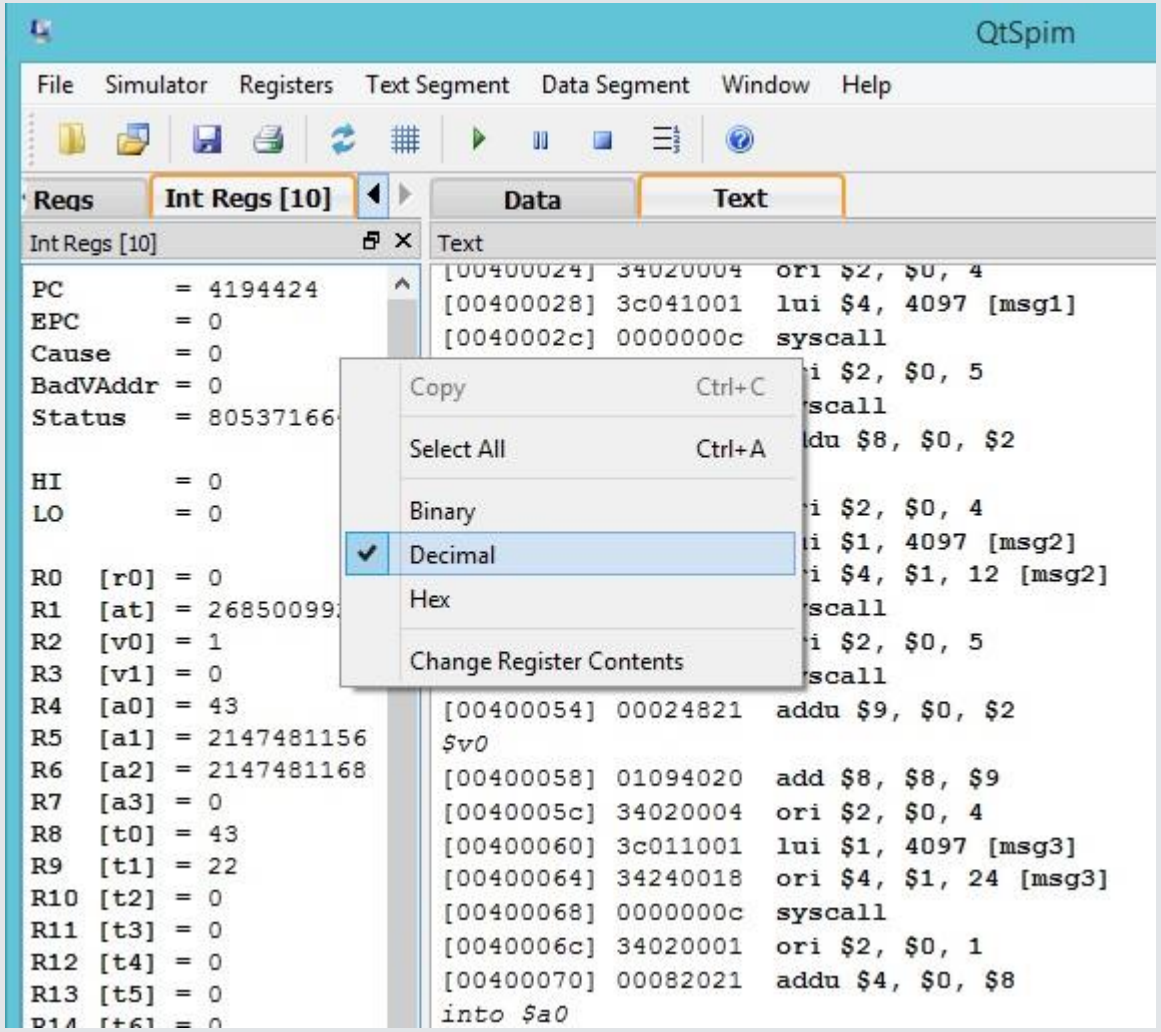


- If a program performs input and/or output, it will be directed to the Console window as shown.

Register Pane



Decimal Representation



Directive

- **.DATA** directive
  - Defines the data segment of a program containing data
  - The program's variables should be defined under this directive
  - Assembler will allocate and initialize the storage of variables
- **.TEXT** directive
  - Defines the code segment of a program containing instructions
- **.GLOBL** directive
  - Declares a symbol as global
  - Global symbols can be referenced from other files
  - We use this directive to declare *main* procedure of a program

Sample Program

```
# Program to print a string on the screen

.data
msg:    .asciiz    “This is CO Lab"

.text
.globl    main

main:          li  $v0, 4                    #System call code for print string
               la  $a0, msg1                 #Address of NULL terminated string
               syscall

               li  $v0, 10                #System call code for exit
               syscall
```

Basic Printing and Reading Exercise

- Write a MIPS program to print Computer Organization on the console
- Write a MIPS program to read an integer from the keyboard and display in the screen.
- Write a MIPS program to read a string from the keyboard and display in the screen.

Load and Store Exercise

- Write a MIPS program to load two numbers from memory and add them.

- Write a MIPS program to load two numbers stored in consecutive memory locations and add them. Store the result in the next memory location.

Thank You