**ARITHMETIC AND LOGICAL OPERATIONS IN QTSPIM(ASSEMBLY LANGUAGE)**

**LAB # 01**



**Fall 2023**

**CSE-304L Computer Organization and Architecture Lab**

Submitted by: **Ali Asghar**

Registration No.: **21PWCSE2059**

Class Section: **C**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Submitted to:

**Dr. Bilal Habib**

Date:

**5th October 2023**

**Department of Computer Systems Engineering**

**University of Engineering and Technology, Peshawar**

**ASSESSMENT RUBRICS COA LABS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **LAB REPORT ASSESSMENT** | | | | |
| **Criteria** | **Excellent** | **Average** | **Nill** | **Marks Obtained** |
| 1. **Objectives of Lab** | All objectives of lab are properly covered  [Marks 10] | Objectives of lab are partially covered  [Marks 5] | Objectives of lab are not shown  [Marks 0] |  |
| 1. **MIPS instructions with**   **Comments and proper indentations.** | All the instructions are well written with comments explaining the code and properly indented  [Marks 20] | Some instructions are missing are poorly commented code  [Marks 10] | The instructions are not properly written  [Marks 0] |  |
| 1. **Simulation run without error and warnings** | The code is running in the simulator without any error and warnings  [Marks 10] | The code is running but with some warnings or errors.  [Marks 5] | The code is written but not running due to errors  [Marks 0] |  |
| 1. **Procedure** | All the instructions are written with proper procedure  [Marks 20] | Some steps are missing  [Marks 10] | steps are totally missing  [Marks 0] |  |
| 1. **OUTPUT** | Proper output of the code written in assembly  [Marks 20] | Some of the outputs are missing  [Marks 10] | No or wrong output  [Marks 0] |  |
| 1. **Conclusion** | Conclusion about the lab is shown and written  [Marks 20] | Conclusion about the lab is partially shown  [Marks 10] | Conclusion about the lab is not shown[Marks0]  [Marks 0] |  |
| 1. **Cheating** |  |  | Any kind of cheating will lead to 0 Marks |  |
| Total Marks Obtained:\_\_\_\_\_\_\_\_\_\_  Instructor Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | | | |

**Task 1:**

Write an assembly language program which takes two numbers from user and add them and show the result on console.

**Code:**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Output:**

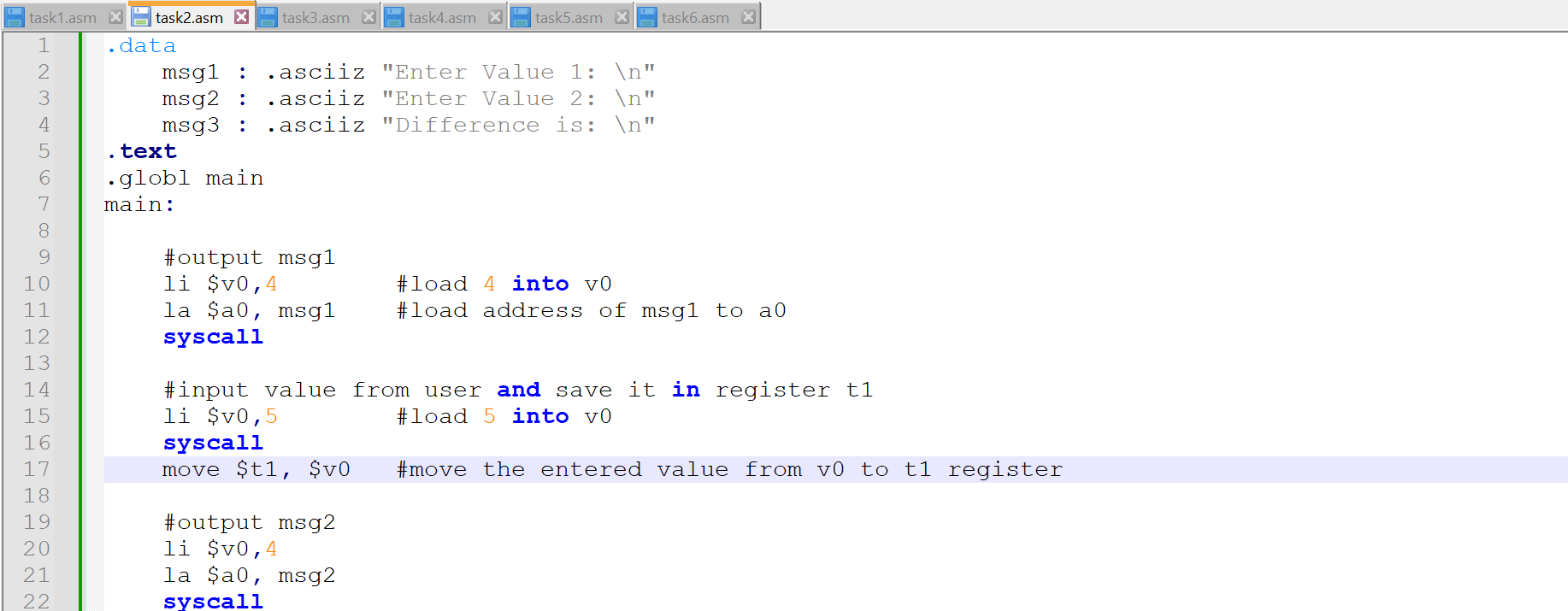
A screenshot of a computer

Description automatically generated

**Task 2:**

Write an assembly language program which takes two numbers from user and subtract them and show the result on console.

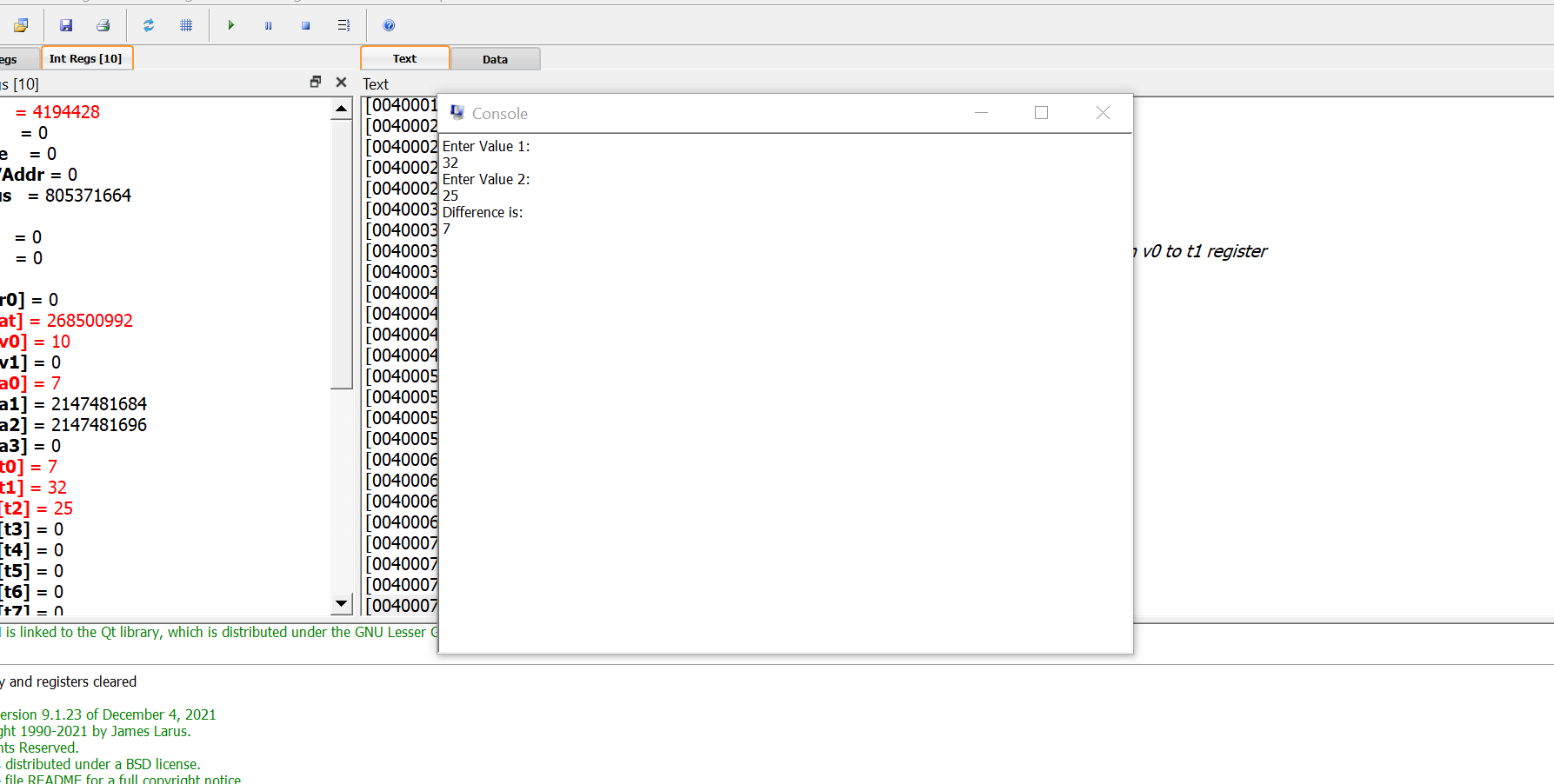
**Code:**



**A screenshot of a computer

Description automatically generated**

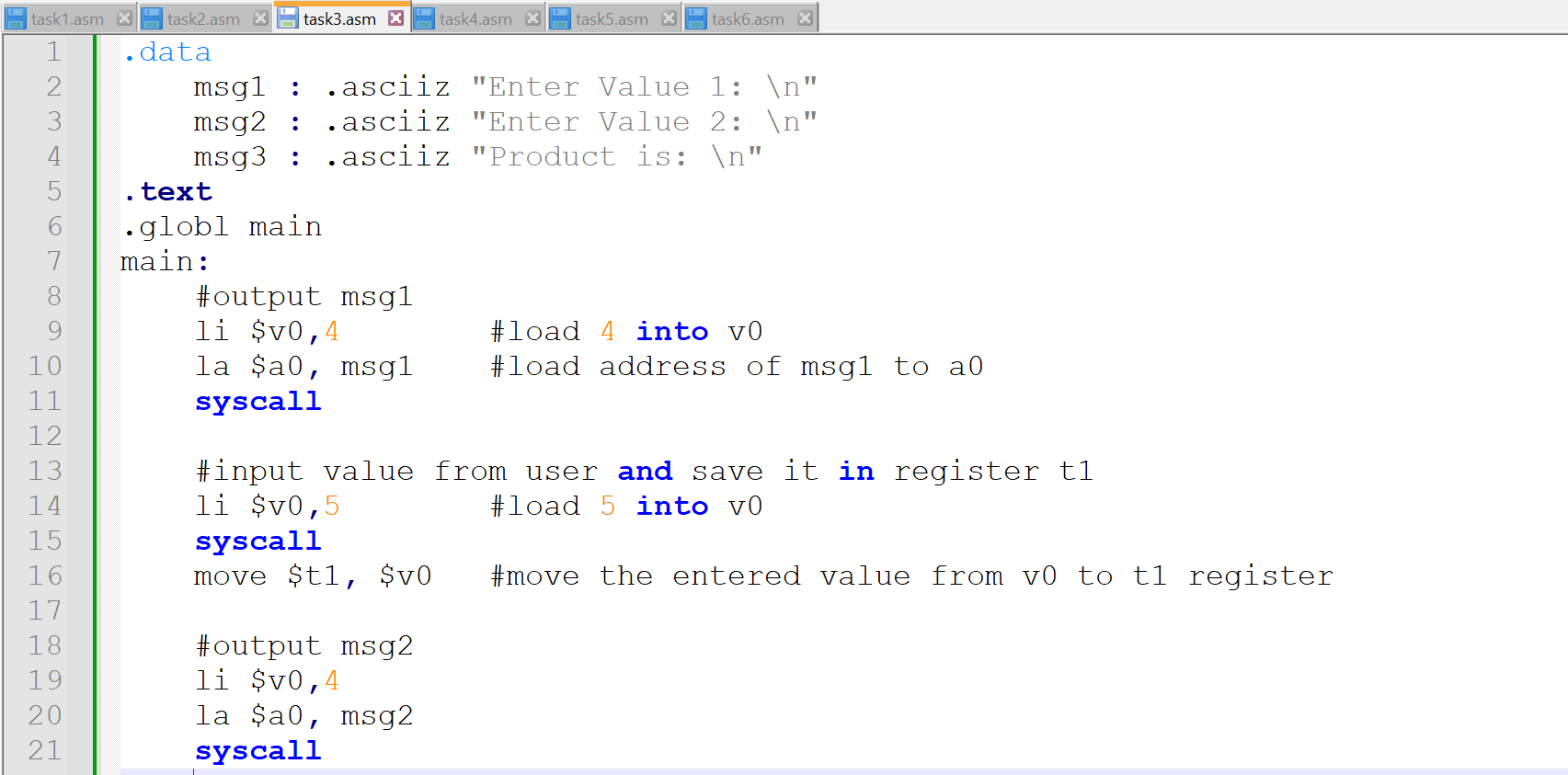
**Output:**



**Task 3:**

Write an assembly language program which takes two numbers from user and multiply them and show the result on console.

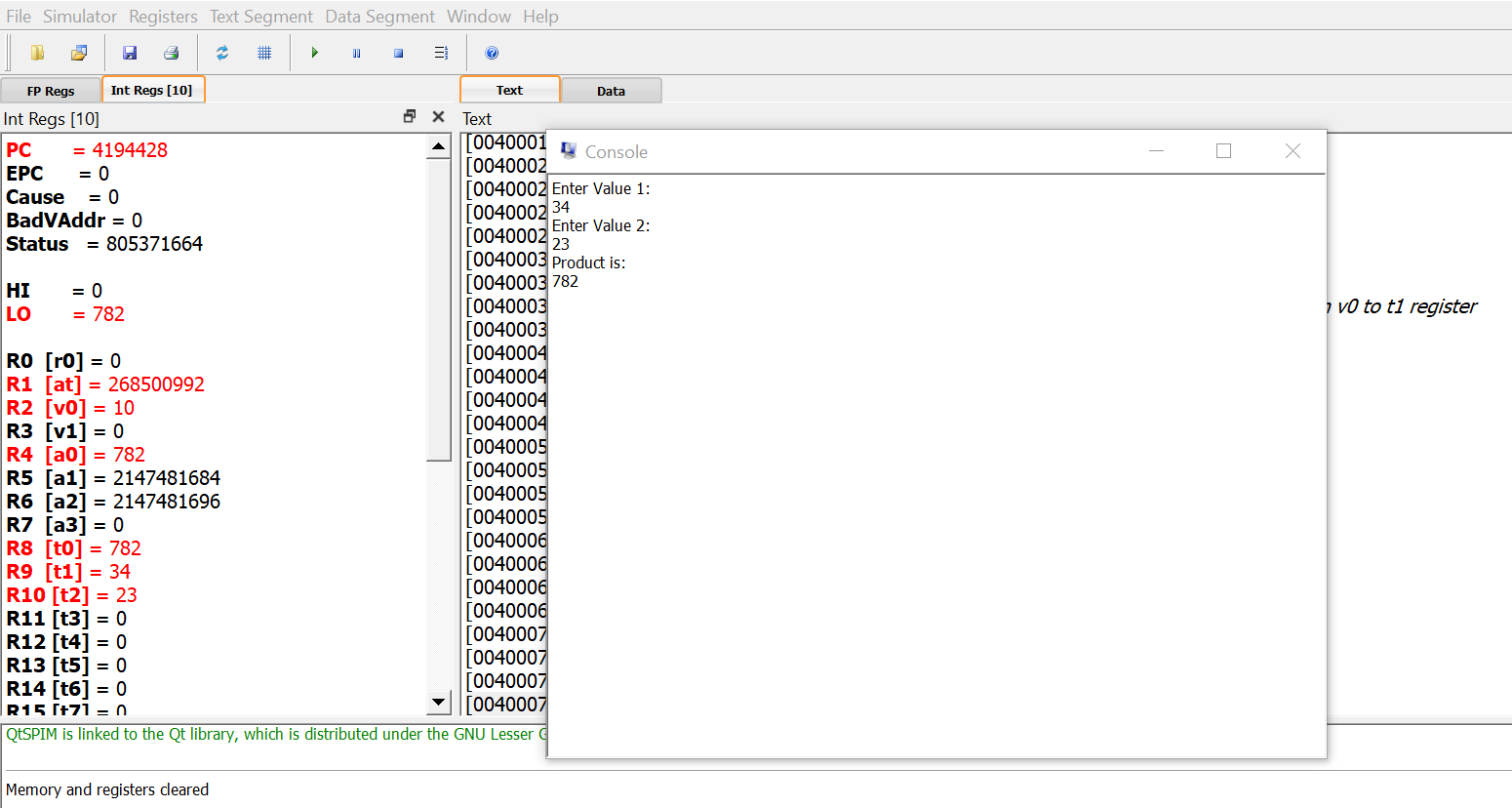
**Code:**

****

**A screenshot of a computer

Description automatically generated**

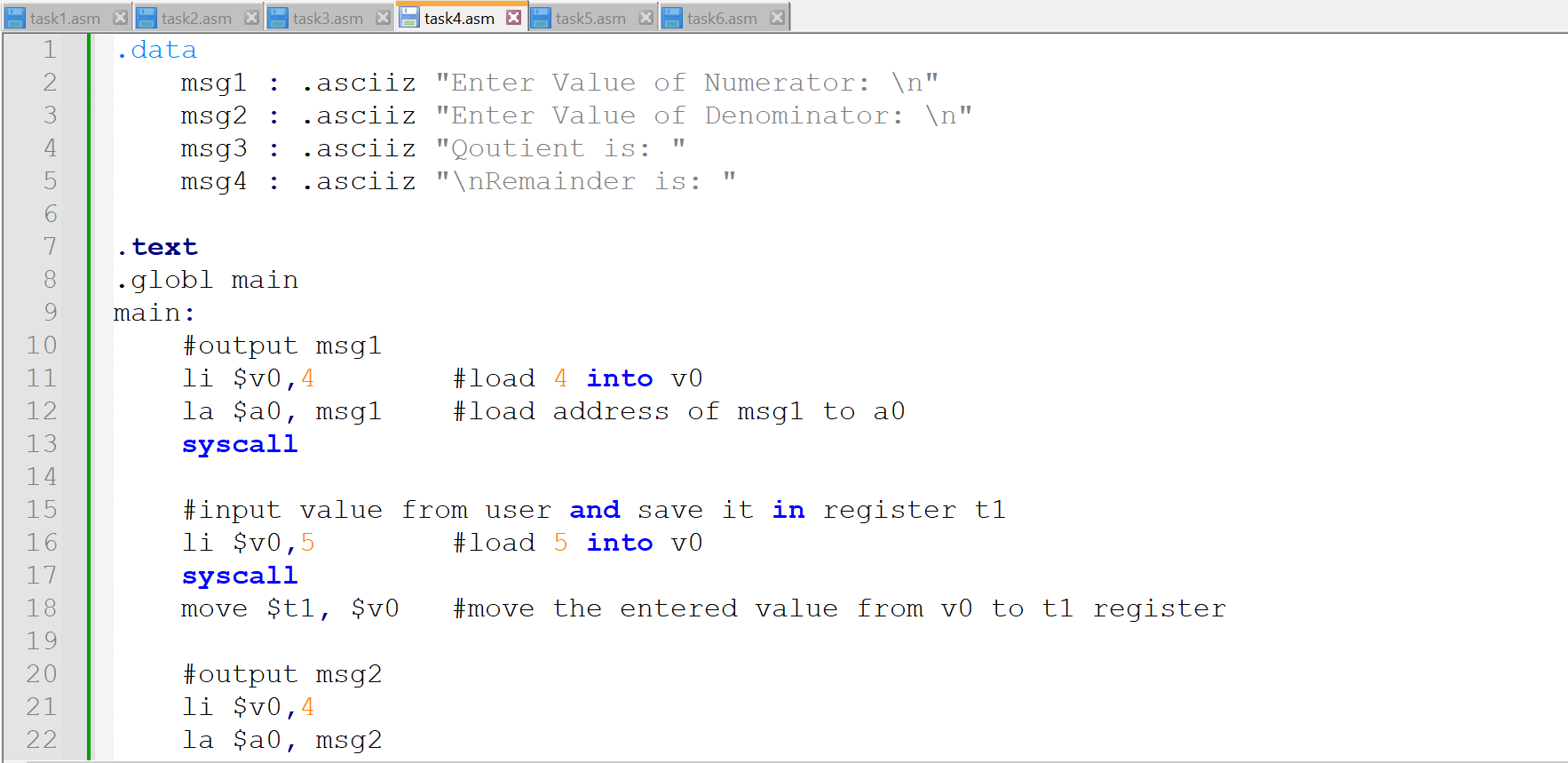
**Output:**

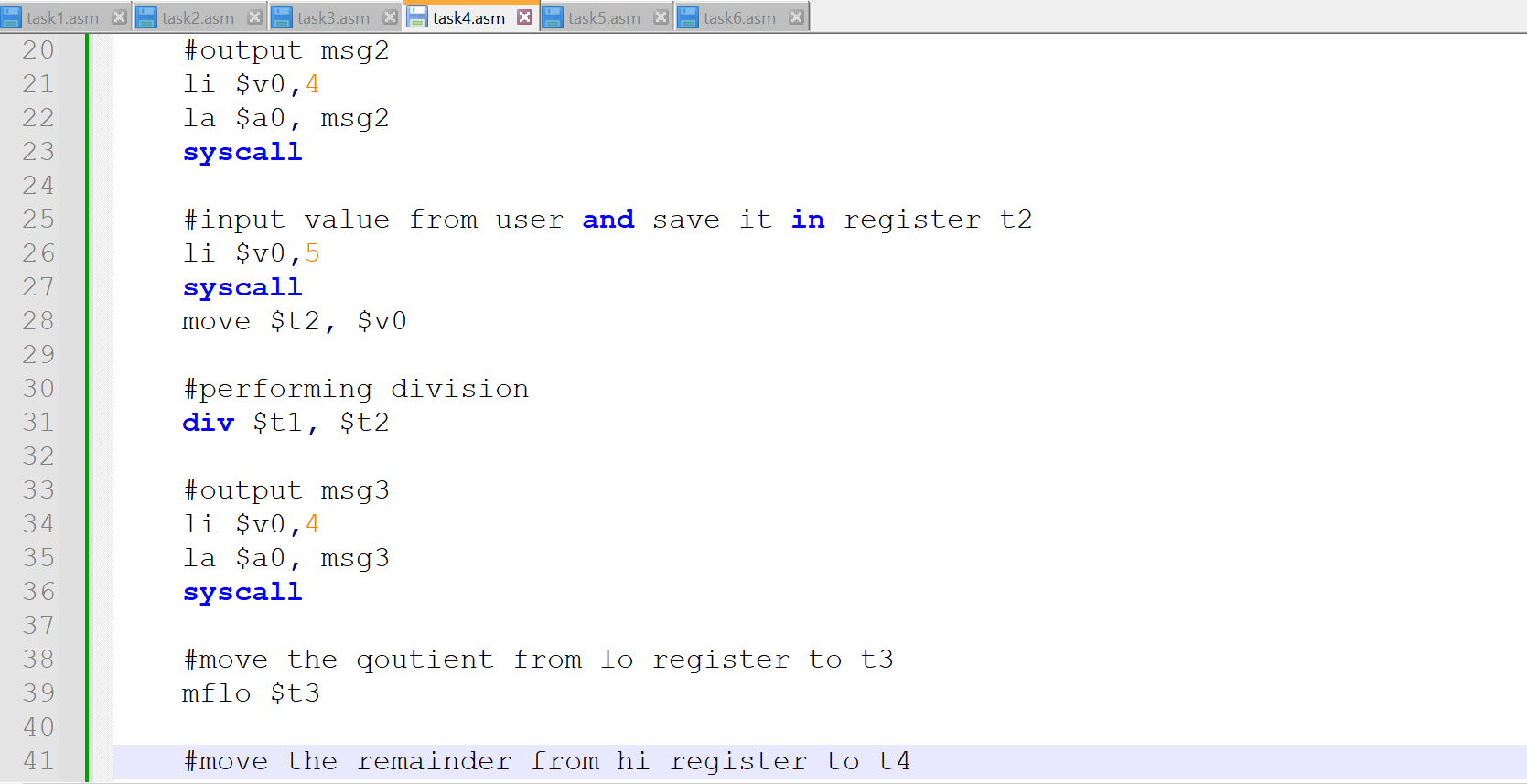


**Task 4:**

Write an assembly language program which takes two numbers from user and divide them and show the result on console.

**Code:**

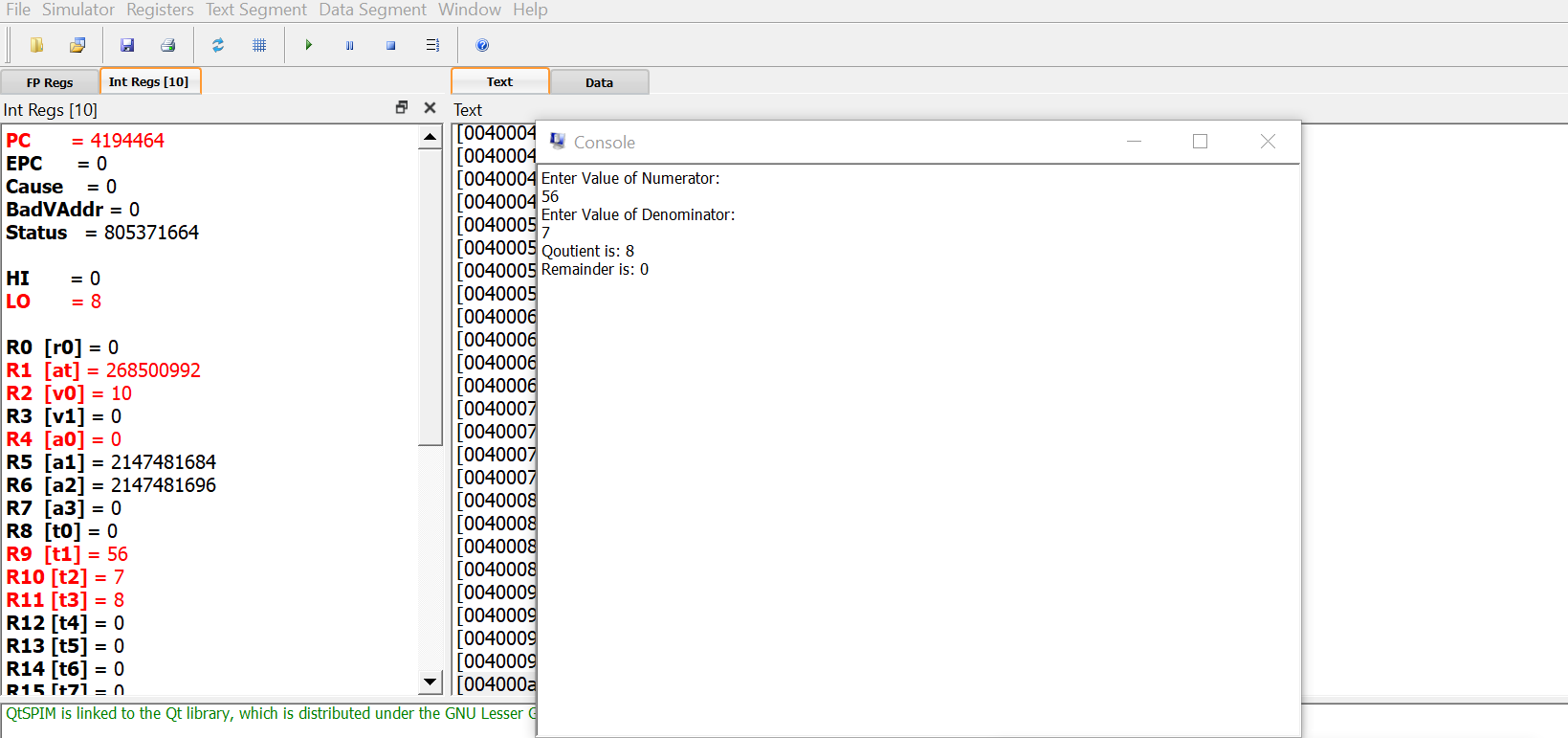
****

****

**A screenshot of a computer screen

Description automatically generated**

**Output:**



**Task 5:**

Write assembly program to multiply two numbers using MULT and extract the bit from high and low registers to general purpose registers.

**Code:**

**A screenshot of a computer

Description automatically generated**

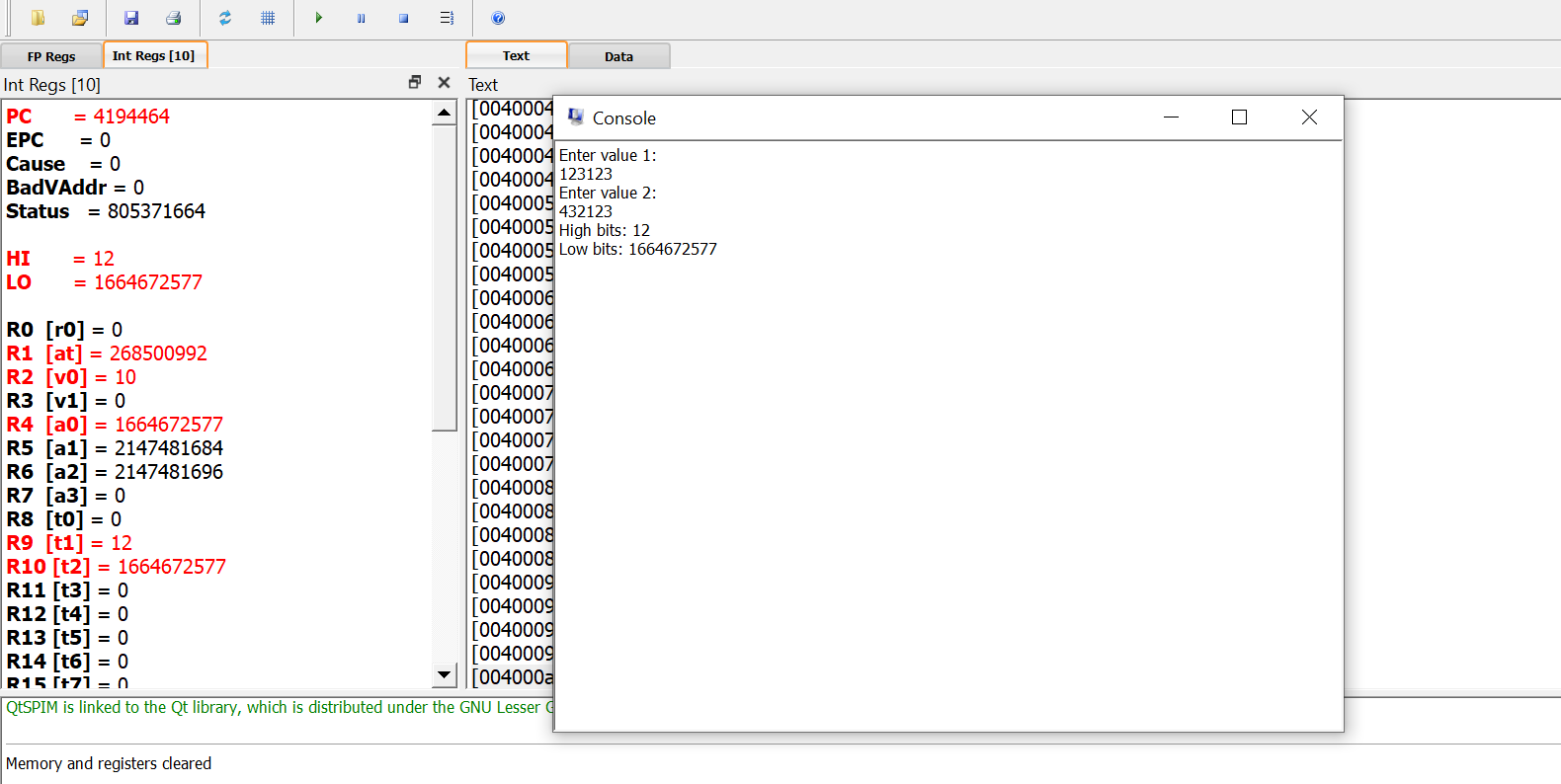
**A screenshot of a computer

Description automatically generated**

**A computer screen shot of a white background

Description automatically generated**

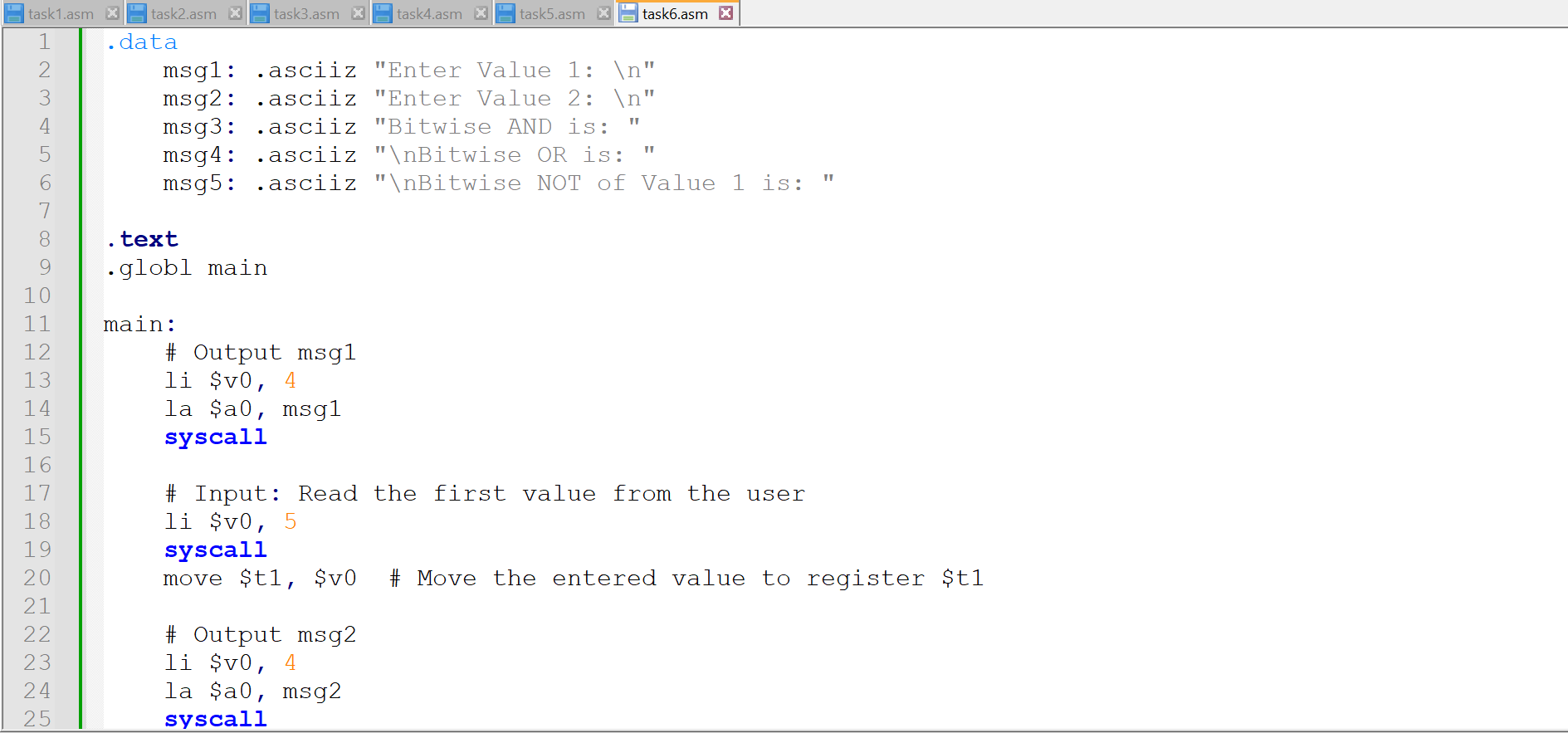
**Output:**

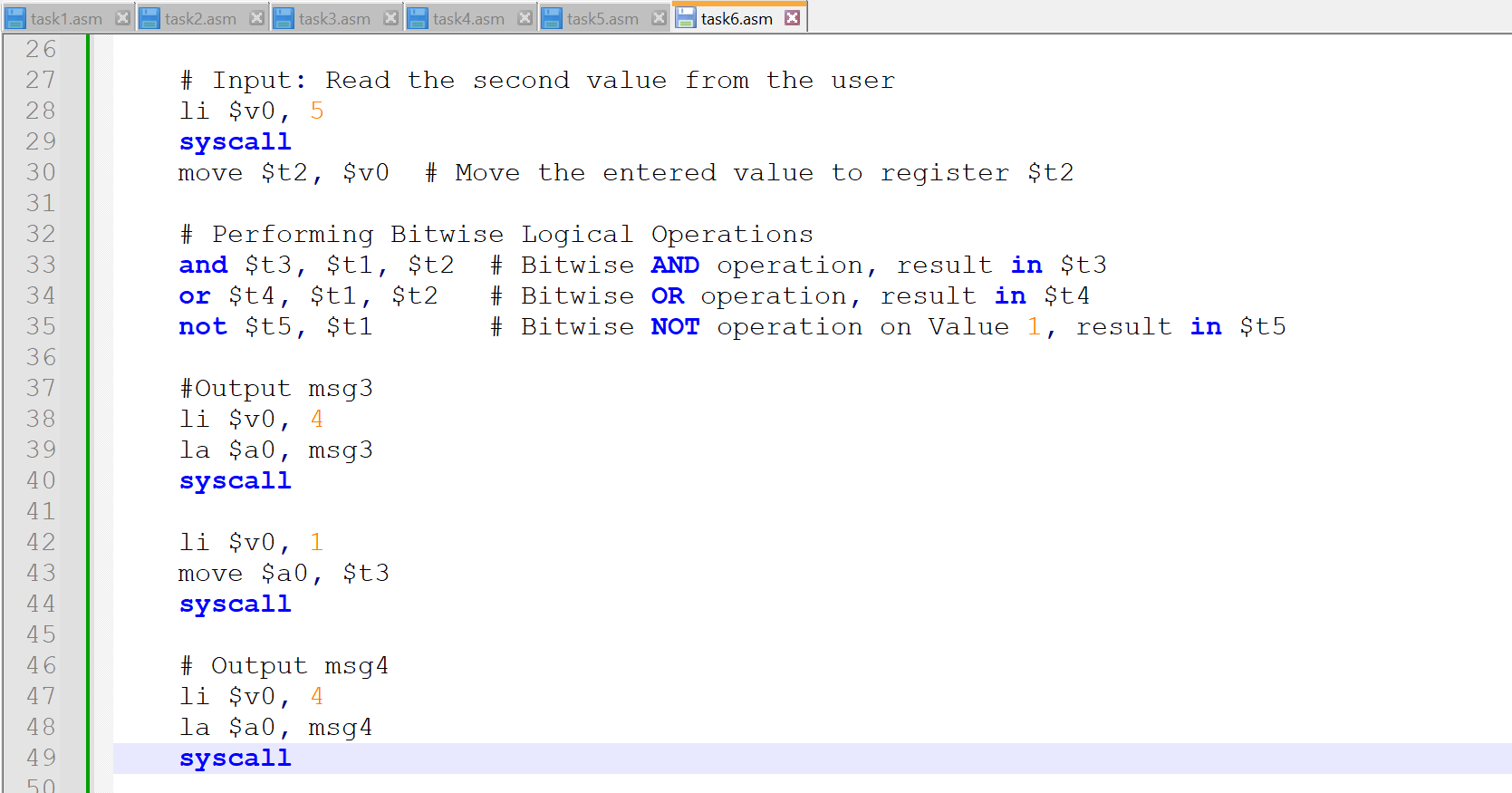


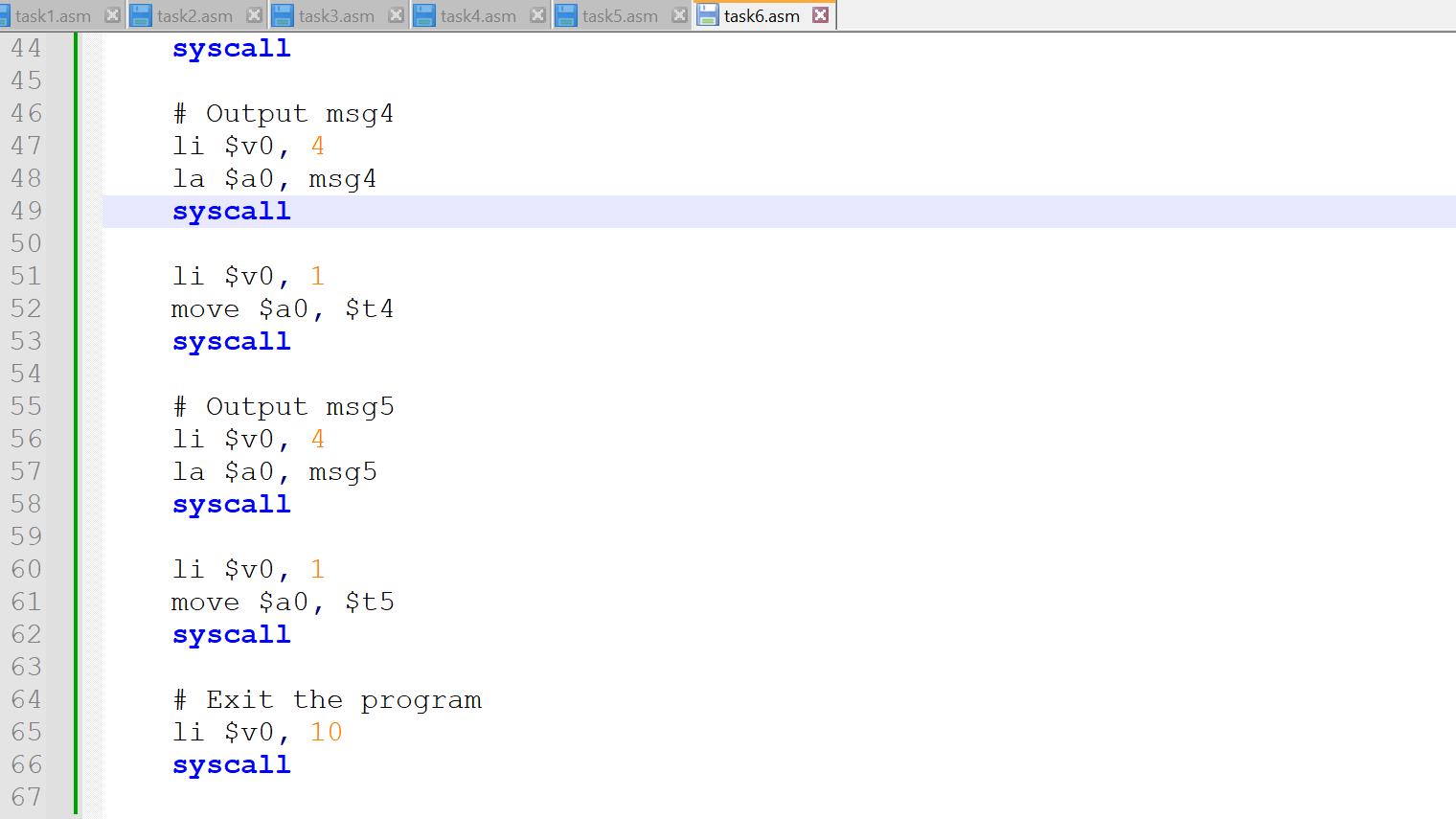
**Task 6:**

Write program to perform AND, OR , NOT operations in MIPS.

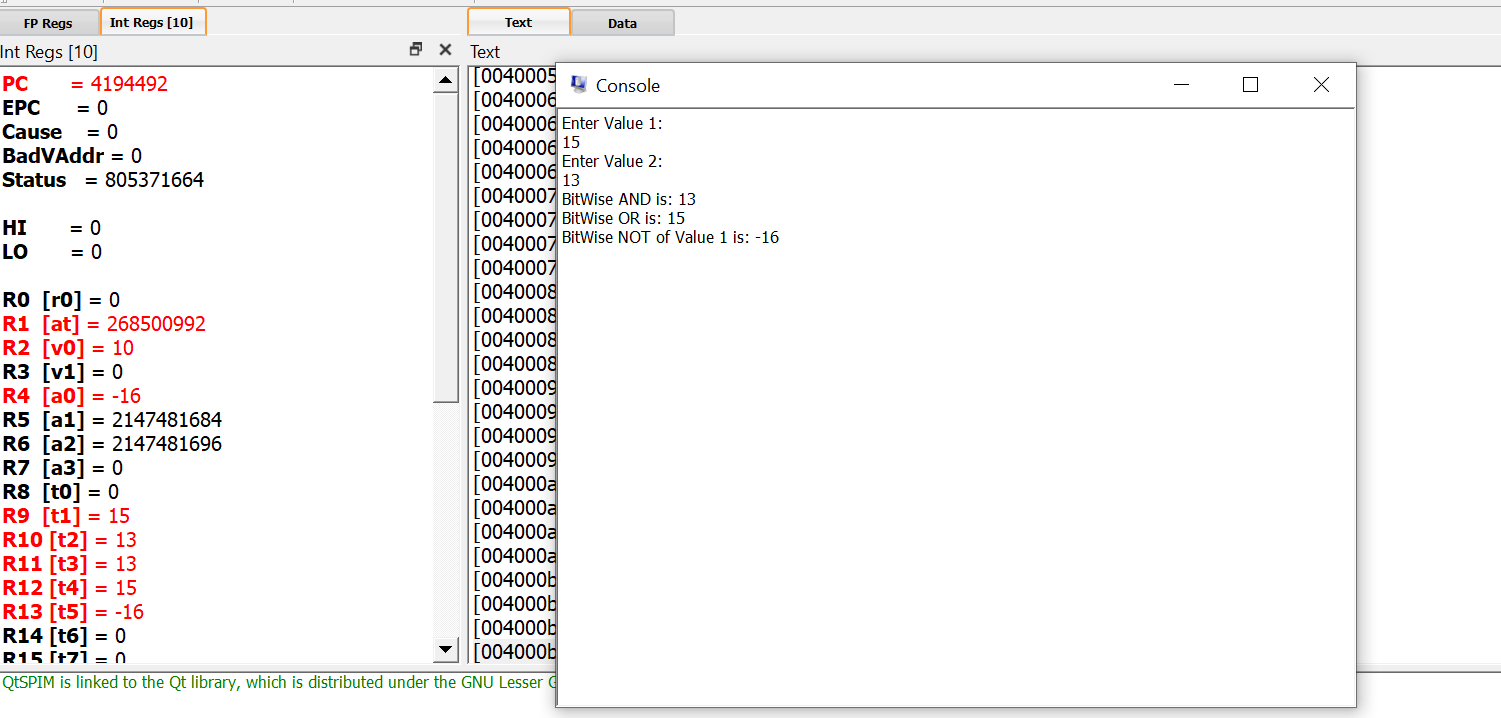
**Code:**

****

****

****

**Output:**



**Conclusion:**

In this lab, I explored basic arithmetic operations (addition, subtraction, multiplication, and division) and logical operations (AND, OR, NOT) in MIPS assembly language. The programs were designed to take input from the user, perform the specified operations, and display the results on the console.