

VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING



COMPUTER ARCHITECTURE - CO2007

ASSIGNMENT

CALCULATOR

LÊ MINH GIA BẢO – 2052396

CC01

HO CHI MINH CITY, 8 MAY 2023
ASSIGNMENT SPECIFICATIONS

1. Introduction

A calculator is typically a portable electronic device used to perform calculations, ranging from basic arithmetic to complex mathematics.

The calculator idea starts from long ago where people would use items in their possession to count. After some time, when civilization became more and more advanced, mechanical calculators emerged and came to be one of the fundamental tools for a person. At the time, such tools can only offer very few functionalities, including addition and subtraction. After the invention of the first solid-state electronic calculator in the 1960s, calculators began to have memory storing capabilities. Therefore, complex instructions can be calculated. Nowadays, calculators are inseparable from the modern society. From elementary schools to university and marketplaces, these pocket-size devices can be found supporting people daily calculations anywhere in the world.

A simple modern calculator should be able to solve basic arithmetic such as addition, subtraction, multiplication, division, and utilizing memory. Some other frequently used functions include factorization, finding Lowest Common Multiple (LCM), Greatest Common Divisor (GCD), square roots, exponents, and converting bases.

2. Implementation Details

2.1. Input Handling

The calculator program begins by prompting users to enter their mathematical expressions through a simple command-line interface. User inputs are read as strings, which are then parsed into recognizable tokens representing numbers and operators. This method ensures that the input process is both efficient and less prone to errors, as the program checks for any invalid characters and syntax errors before proceeding with any calculations.

2.2. Mathematical Operations

- Addition, Subtraction, Multiplication, and Division: These basic operations handle both integer and floating-point numbers.
- Exponentiation: Allows users to perform power calculations.
- Factorials: Implemented for integer inputs to calculate the product of all positive integers up to the given number.
- Parentheses: The program respects the mathematical precedence by evaluating expressions within parentheses first.

2.3. Error Handling

Error handling is a critical component of the calculator. The program identifies several potential input errors, such as non-numeric characters, misplaced operators, and unmatched parentheses. Upon detection, it provides descriptive error messages to the user without crashing, allowing for corrective actions to be made.

3. Program Structure and Functionality

The program starts by setting up an infinite loop, which continuously prompts the user to enter mathematical expressions. The key MIPS assembly instructions involved in this process are:

Parsing and Validation:

The input string is then checked for valid characters and balanced parentheses before any computation is attempted. This is done through a series of conditional checks and branching:

- **Character validation** ensures that only numbers, operators, and specific symbols (like parentheses) are entered.
- **Parentheses counting** ensures that every opening parenthesis has a matching closing parenthesis.

Mathematical Operations:

Each operation such as addition, subtraction, multiplication, and division has a specific section of code:

- **Addition (Add):** Accumulates values into a register by summing them up.
- **Subtraction (Sub):** Subtracts the next value from the accumulated sum.
- **Multiplication (Mul):** Multiplies the accumulated result by the next value.
- **Division (Div):** Divides the current result by the next value.

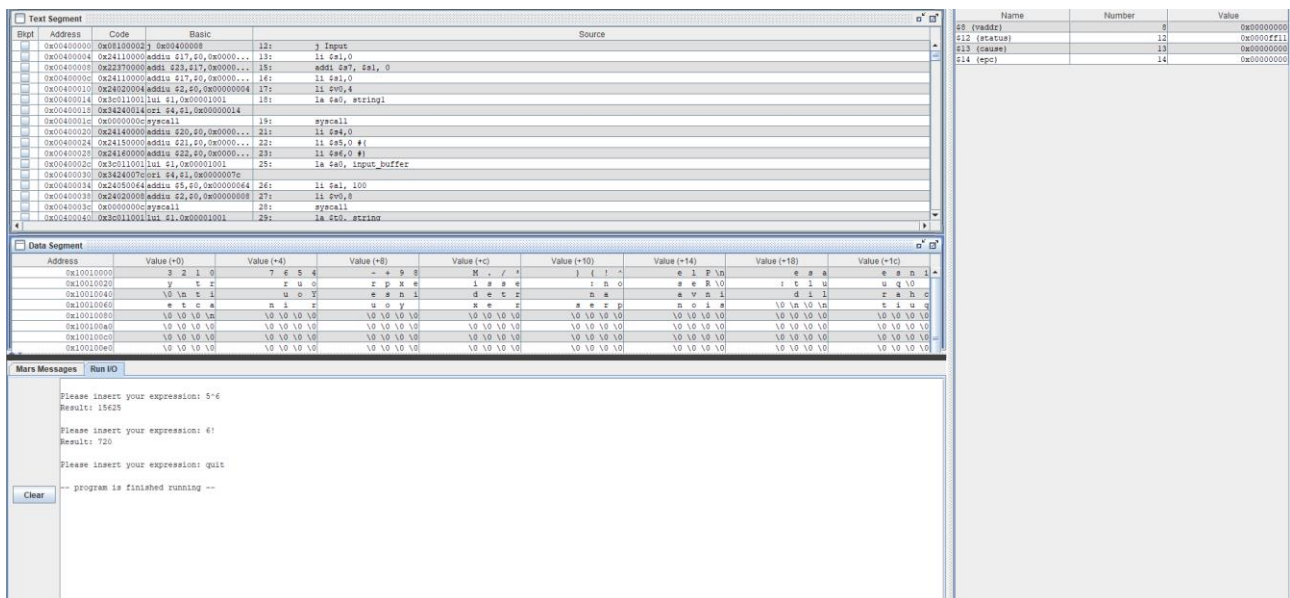
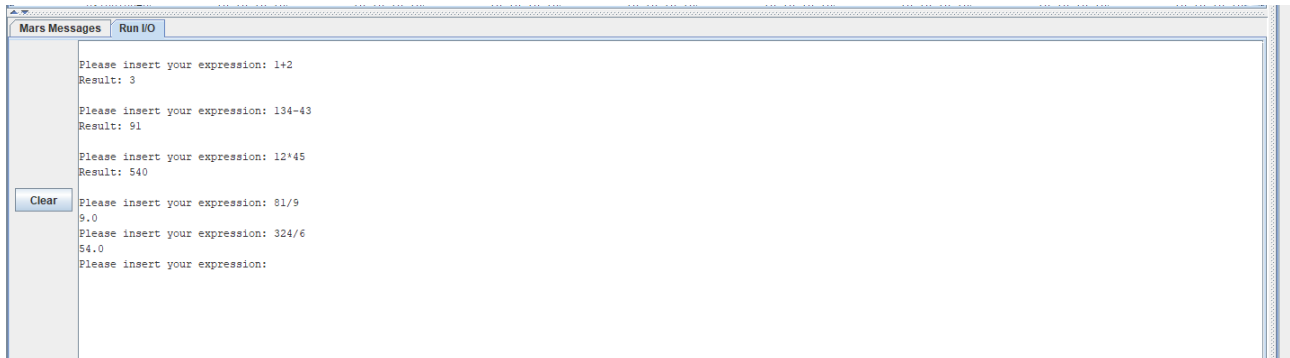
Error Handling:

Error handling is integrated into the input validation phase and during each mathematical operation:

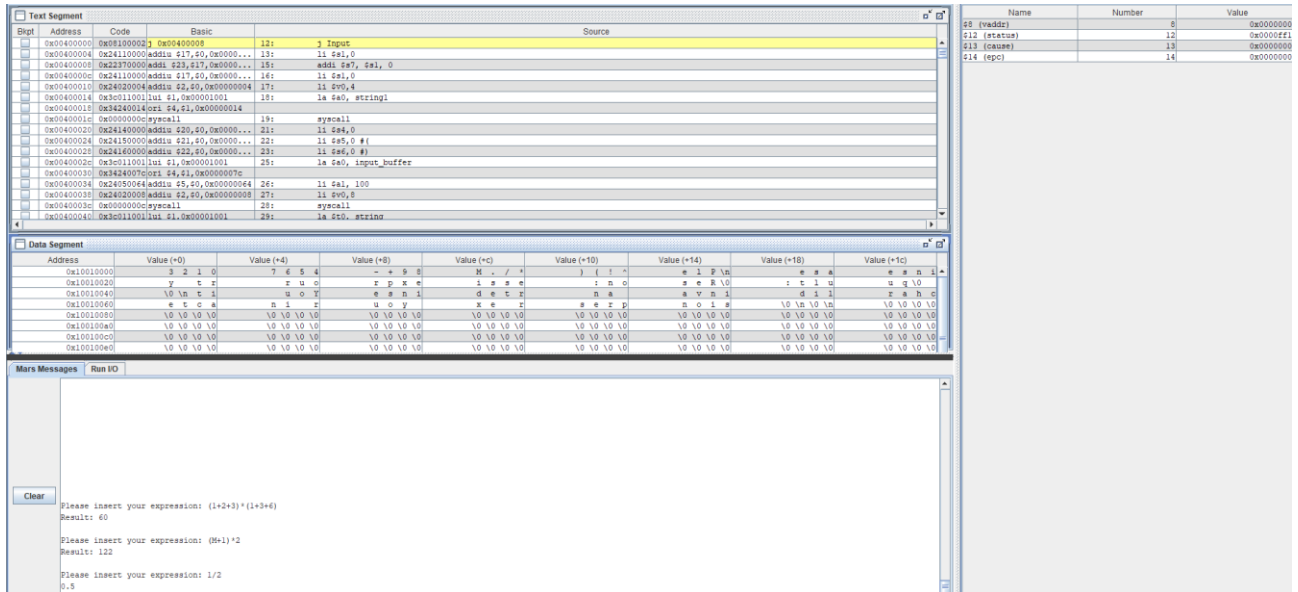
- Invalid character detection immediately notifies the user and asks for re-entry of the expression.
- Mathematical errors like division by zero are checked before performing the division.

4. Testing and Results

4.1. Basic Operations:



4.2. Complex Expressions:

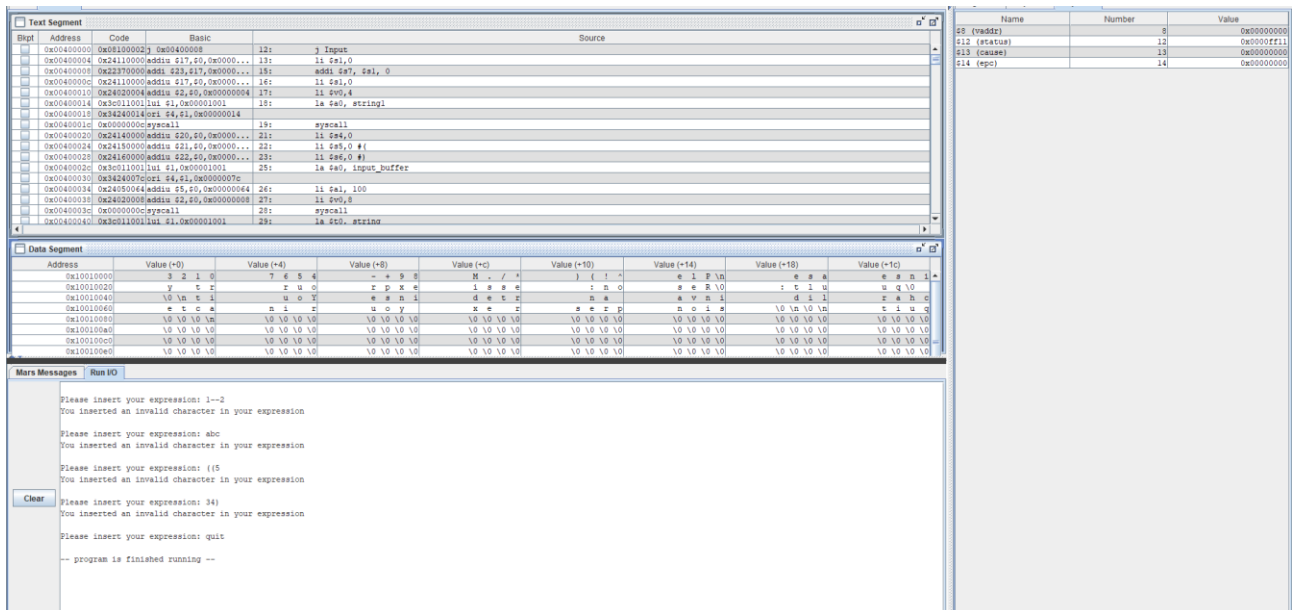


The screenshot shows the Mars IDE interface. The top window displays assembly code for the 'Text Segment'. The middle window shows the 'Data Segment' with various memory addresses and values. The bottom window, 'Mars Messages', shows the results of several complex expressions entered in the 'Run IO' tab:

- Please insert your expression: $(1+2+3)*(1+4+6)$
Result: 60
- Please insert your expression: $(H+1)^2$
Result: 122
- Please insert your expression: $1/2$
Result: 0.5

Can process decimal number

4.3. Error Handling:



The screenshot shows the Mars IDE interface. The bottom window, 'Mars Messages', shows the results of several expressions entered in the 'Run IO' tab, including error messages:

- Please insert your expression: $1-2$
You inserted an invalid character in your expression
- Please insert your expression: abc
You inserted an invalid character in your expression
- Please insert your expression: $(15$
You inserted an invalid character in your expression
- Please insert your expression: 34)
You inserted an invalid character in your expression
- Please insert your expression: quit
-- program is finished running --

Also a “quit” command to exit the program, also possibility to save the result of the last calculation into the memory.

5. Challenges

The missing part in my program is can not process to calculate the equation that priority to calculate multiply and divide before calculate add and subtract, it is just follow from left to right.

And also can not create a clearly log file to see all the input from user.

6. Conclusion

The project of implementing an interactive calculator in MIPS assembly language provided a deep dive into the challenges and intricacies of low-level programming. By successfully handling basic arithmetic operations, complex mathematical functions like exponentiation and factorial, and robust error detection, the calculator has become a comprehensive tool for performing a wide range of mathematical calculations although there are some features need to be finished such as log files and priority calculation.

—————**END**—————