

CCCS-217

Computer organization and architecture

TS2

Basic calculator

Group 5

Group members:

Abdulhadi Bandar Alghamdi 2340745

Rakan Fawuzi Jalalah 2340267

Mohannad Abdullah Alsherhri 2340515

10910-TS2



To make our calculator more creative we adjusted couple of things

For example, after the user chooses an operation, we adjusted the result message to be more specific for each operation and it will also print the numbers that the user entered, as shown in the pictures below:

```
The result of adding 12 to 15 is: 27

The result of subtracting 20 from 12 is: 8

The result of multiplying 40 by 2 is: 80

The result of dividing 100 by 4 is: 25
```

Also, after the user gets the result, we added the options to let the user do another arithmetic operation or exit the program by writing 1 or 0 as true or false, as shown in the picture before:

```
Do you want to perform another operation?
Enter 1 to continue or 0 to exit:
```

Also, it is allowed mid choosing the operation list just if the user wants to exit the program

```
Please select the desired operation:
1. Addition
2. Subtraction
3. Multiplication
4. Division
0. Exit
```

If the user decides to exit the program it will also print a " thank you " message as shown:

```
Do you want to perform another operation?
Enter 1 to continue or 0 to exit: 0
Thank you for using our simple assembly language calculator.
```

And of course, if the user wants to do another operation it will continue starting all over again as shown

```
Do you want to perform another operation?
Enter 1 to continue or 0 to exit: 1
Enter the first number please:
```

Also, we printed a specific message if a user tries to divide any number by 0 as shown

```
Enter the first number please: 15
Enter the second number please: 0

Please select the desired operation:
1. Addition
2. Subtraction
3. Multiplication
4. Division
0. Exit
4

Error: Division by zero is not allowed.
```

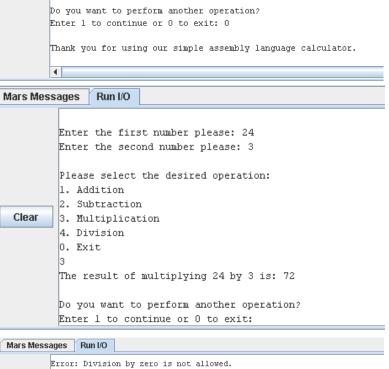
Here is a screenshot of the full code and some testing with different arithmetic operations

```
Edit Execute
project0.txt*
                      .asciiz "Enter the first number please:
                     .asciiz "Enter the second number please: "
.asciiz "\nPlease select the desired operation:\nl. Addition\n2. Subtraction\n3. Multiplication\n4. Division\n0. Exit\n"
         msg2:
        menu:
        result_add: .asciiz "The result of adding "
        result sub: .asciiz "The result of subtracting" result mul: .asciiz "The result of multiplying"
         result_div: .asciiz "The result of dividing "
                    .asciiz " by "
         by_msg:
                      .asciiz " to "
10
         to msg:
         from_msg: .asciiz " from "
         is_msg: .asciiz "is: "
error_msg: .asciiz "\nError: Division by zero is not allowed.\n"
12
13
         continue_msg: .asciiz "\nDo you want to perform another operation?\nEnter 1 to continue or 0 to exit: "
         exit msg: .asciiz "\nThank you for using our simple assembly language calculator.\n"
newline: .asciiz "\n"
1.5
16
18 .text
19 main:
20 restart:
      # Printing the first message
li $v0, 4 # Syscal
21
        li $v0, 4 # Syscall for printing a string
la $a0, msgl # Load message address
23
24
       syscall
25
        # Reading & Storing the first number
26
                             # Syscall for reading an integer
28
         syscall
29
        move $t0, $v0
                               # Store first number in $t0
         # Printing the second message
31
                              # Syscall for printing a string
         li $v0, 4
         la $aO, msg2
         syscall
```

```
Edit Execute
 project0.txt*
 52
         beq $t2, 1, addition
 53
         beq $t2, 2, subtraction
beq $t2, 3, multiplication
 54
         beq $t2, 4, division
 56
         beq $t2, 0, exit_program
 57
 58
         # Default case if the user's input is invalid it will restart
 59
         j restart
 60
 61 addition:
         add $t3, $t0, $t1 # Add $t0 and $t1, store in $t3
 62
         la $aO, result_add
 64
         j print_result
 65
        sub $t3, $t0, $t1 # Subtract $t1 from $t0, store in $t3
 67
         la $aO, result_sub
 68
 69
         j print_result
 70
 71 multiplication:
 72
         73
         la $aO, result_mul
 74
         j print_result
 75
 76 division:
         # Check for division by zero
 78
79
         beq $tl, $zero, div_error
         div $t0, $t1  # Divide $t0 by $t1
mflo $t3  # Store quotient in $t3
        la $a0, result_div
j print_result
 81
 82
 84 div error:
        # Print error message for division by zero
 85
 86
         li $v0, 4
         la $a0, error_msg
 87
 88
 89
         j restart
 90
 91 continue_prompt:
 92
        # Ask to continue or exit
         li $v0, 4
 93
         la $a0, continue_msg
 95
         syscall
 96
         # Read user decision
                      # Syscall for reading an integer
 98
         li $v0, 5
 99
 100
         beq $v0, 1, restart # If 1, restart
Line: 147 Column: 54 🗹 Show Line Numbers
```

```
project0.txt*
        beq $v0, 1, restart # If 1, restart
beq $v0, 0, exit_program # If 0, exit
100
102
        i restart
                          # Default: restart
103
104 print_result:
       # Print operation message
105
106
        li $v0, 4
                          # Syscall for printing a string
107
        syscall
108
        109
110
        move $a0, $t0
        svscall
113
114
        # Print "from" for subtraction or "by/to" for others
        beq $t2, 2, print_from
116
        beq $t2, 1, print_to
117
        la $aO, by msg
118
        j print_second
119
120 print_from:
121
        la $a0, from_msg
122
        j print_second
124 print_to:
       la $aO, to_msg
126
127 print second:
       # Print "from", "by", or "to"
129
        li $v0, 4
        syscall
130
        # Print second number
132
```

```
146
       # Print newline for more clean looking interface
147
148
        li $v0, 4
149
        la $a0, newline
150
152
        i continue prompt
153
155
        # Print exit message
        li $v0, 4
156
157
        la $a0, exit_msg
        syscall
158
160
        # Exit program
        li $v0, 10
                         # Syscall for exit
161
        syscall
163
4
Line: 147 Column: 54 🗹 Show Line Numbers
                                                                                          Mars Messages Run I/O
 Mars Messages Run I/O
                                                                                                     Enter the first number please: 7
           Enter the first number please: 12
                                                                                                     Enter the second number please: 15
           Enter the second number please: 13
                                                                                                     Please select the desired operation:
           Please select the desired operation:
                                                                                                     1. Addition
           1. Addition
                                                                                                     2. Subtraction
           2. Subtraction
                                                                                                     3. Multiplication
           3. Multiplication
                                                                                                     4. Division
            4. Division
                                                                                                     0. Exit
           0. Exit
   Clear
                                                                                            Clear
           The result of adding 12 to 13 is: 25
                                                                                                     The result of subtracting 7 from 15 is: -8
```



Enter the first number please: 100

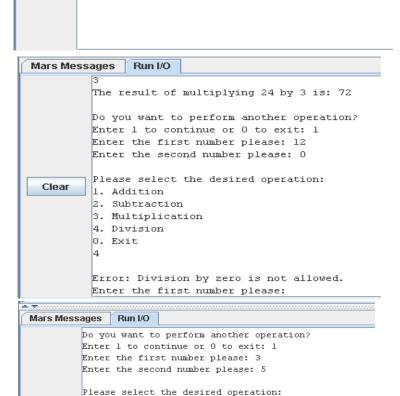
Enter the second number please: 4

Addition
 Subtraction

4. Division

Multiplication

Please select the desired operation:



1. Addition

Carber

Do you want to perform another operation?

Enter 1 to continue or 0 to exit: