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*

```
.data
    msg1:       .asciiz "Enter the first number please: "
    msg2:       .asciiz "Enter the second number please: "
    menu:       .asciiz "\nPlease select the desired operation:\n1. Addition\n2. Subtraction\n3. Multiplication\n4. Division\n0. Exit\n"
    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 "
    by_msg:     .asciiz " by "
    to_msg:     .asciiz " to "
    from_msg:   .asciiz " from "
    is_msg:     .asciiz " is: "
    error_msg:  .asciiz "\nError: Division by zero is not allowed.\n"
    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"

.text
main:
restart:
    # Printing the first message
    li $v0, 4           # Syscall for printing a string
    la $a0, msg1        # Load message address
    syscall

    # Reading & Storing the first number
    li $v0, 5           # Syscall for reading an integer
    syscall
    move $t0, $v0       # Store first number in $t0

    # Printing the second message
    li $v0, 4           # Syscall for printing a string
    la $a0, msg2
    syscall
```

Edit | Execute

project0.txt*

```
    beq $t2, 1, addition
    beq $t2, 2, subtraction
    beq $t2, 3, multiplication
    beq $t2, 4, division
    beq $t2, 0, exit_program

    # Default case if the user's input is invalid it will restart
    j restart

addition:
    add $t3, $t0, $t1   # Add $t0 and $t1, store in $t3
    la $a0, result_add
    j print_result

subtraction:
    sub $t3, $t0, $t1   # Subtract $t1 from $t0, store in $t3
    la $a0, result_sub
    j print_result

multiplication:
    mul $t3, $t0, $t1   # Multiply $t0 and $t1, store in $t3
    la $a0, result_mul
    j print_result

division:
    # Check for division by zero
    beq $t1, $zero, div_error
    div $t0, $t1        # Divide $t0 by $t1
    mflo $t3            # Store quotient in $t3
    la $a0, result_div
    j print_result

div_error:
    # Print error message for division by zero
    li $v0, 4
    la $a0, error_msg
    syscall
    j restart

continue_prompt:
    # Ask to continue or exit
    li $v0, 4
    la $a0, continue_msg
    syscall

    # Read user decision
    li $v0, 5           # Syscall for reading an integer
    syscall
    beq $v0, 1, restart # If 1, restart
```

Line: 147 Column: 54 ☑ Show Line Numbers

Edit | Execute

project0.txt*

```
    beq $v0, 1, restart # If 1, restart
    beq $v0, 0, exit_program # If 0, exit
    j restart           # Default: restart

print_result:
    # Print operation message
    li $v0, 4           # Syscall for printing a string
    syscall

    # Print first number
    li $v0, 1           # Syscall for printing an integer
    move $a0, $t0       # Load first number into $a0
    syscall

    # Print "from" for subtraction or "by/to" for others
    beq $t2, 2, print_from
    beq $t2, 1, print_to
    la $a0, by_msg
    j print_second

print_from:
    la $a0, from_msg
    j print_second

print_to:
    la $a0, to_msg

print_second:
    # Print "from", "by", or "to"
    li $v0, 4
    syscall

    # Print second number
```

```
146
147     # Print newline for more clean looking interface
148     li $v0, 4
149     la $a0, newline
150     syscall
151
152     j continue_prompt
153
154 exit_program:
155     # Print exit message
156     li $v0, 4
157     la $a0, exit_msg
158     syscall
159
160     # Exit program
161     li $v0, 10          # Syscall for exit
162     syscall
163
```

Line: 147 Column: 54 Show Line Numbers

Mars Messages | Run I/O

Clear

```
Enter the first number please: 12
Enter the second number please: 13

Please select the desired operation:
1. Addition
2. Subtraction
3. Multiplication
4. Division
0. Exit
1
The result of adding 12 to 13 is: 25

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.
```

Mars Messages | Run I/O

Clear

```
Enter the first number please: 7
Enter the second number please: 15

Please select the desired operation:
1. Addition
2. Subtraction
3. Multiplication
4. Division
0. Exit
2
The result of subtracting 7 from 15 is: -8

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

Mars Messages | Run I/O

Clear

```
Enter the first number please: 24
Enter the second number please: 3

Please select the desired operation:
1. Addition
2. Subtraction
3. Multiplication
4. Division
0. Exit
3
The result of multiplying 24 by 3 is: 72

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

Mars Messages | Run I/O

Clear

```
3
The result of multiplying 24 by 3 is: 72

Do you want to perform another operation?
Enter 1 to continue or 0 to exit: 1
Enter the first number please: 12
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.
Enter the first number please:
```

Mars Messages | Run I/O

```
Error: Division by zero is not allowed.
Enter the first number please: 100
Enter the second number please: 4

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

Mars Messages | Run I/O

```
Do you want to perform another operation?
Enter 1 to continue or 0 to exit: 1
Enter the first number please: 3
Enter the second number please: 5

Please select the desired operation:
1. Addition
2. Subtraction
```