Matthew Michael Sherlin
Dr. Augustine Samba
Computer Organization
November 6, 2020

Assembly Code File:

```
# Matthew Michael Sherlin
# Dr. Augustine Samba
# Computer Organization
# November 6, 2020

.data
	prompt1: .asciiz "Adding Machine Program\nEnter integer to add to total\nor enter '0' to exit program and\nprint the sum.\n\n"
	prompt2: .asciiz "Enter an integer: "
	message1: .asciiz "\nFinal value of integers: "
	message2: .asciiz "\nEnding program"

.text

li $v0, 4		#reading the first prompt
la $a0, prompt1
syscall

loop:
li $v0, 4		#reading the second prompt
la $a0, prompt2
syscall

li $v0, 5		#entering user input
syscall
move $t0, $v0		#moving user input into temporary register

add $t1, $t1, $t0

bnez $t0, loop		#if integer != 0, continue the loop

li $v0, 4		#reading the first message
la $a0, message1
syscall

li $v0, 1		#displaying total
move $a0, $t1
syscall

li $v0, 4		#reading the second message
la $a0, message2
```

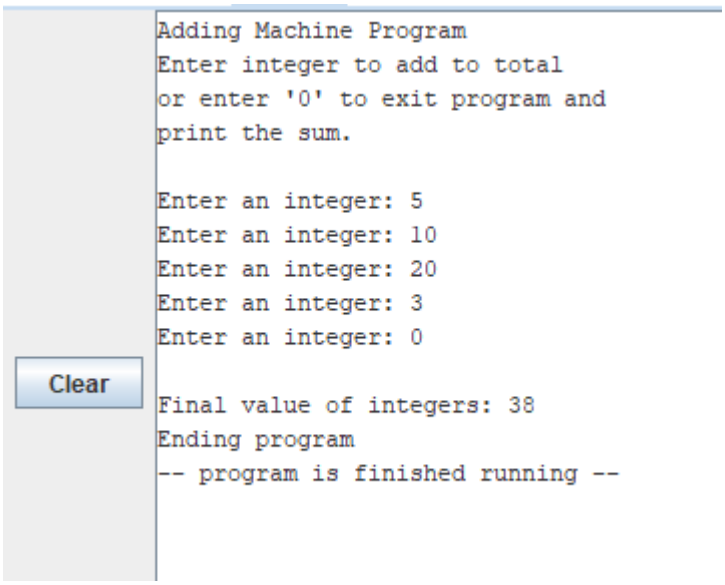
```
syscall

li $v0, 10          #terminate program run and exit
syscall
```

Project Implementation:

In order to get this program to work, I first began by reading the prompt out and then creating a loop label to jump back to. Inside the label, I begin by retrieving the first integer entered by the user. I used basic la and li instructions to achieve this and reading the prompts. Next, I added the integer into a temporary register that will contain the sum by just using the add instruction. After, I used the bnez instruction (branch if not equal to zero) to see if the integer added by the user was zero or not. If it was not, I jumped back to the loop label I created. Otherwise, I read the final message and sum out, and then I ended the program. This program would loop forever until a 0 was entered.

Working Code Screen Print:



Transcription:

Adding Machine Program
Enter integer to add to total
or enter '0' to exit program and
print the sum.

Enter an integer: 5
Enter an integer: 10
Enter an integer: 20
Enter an integer: 3
Enter an integer: 0

Final value of integers: 38
Ending program
-- program is finished running --

Conclusion:

To conclude, I learned a new instruction during this assignment. I was looking for the branch if equal to instruction when I saw the bnez or branch if not equal to zero instruction. This instruction was perfect for the scenario and I used it for this program. Besides learning the new instruction, I didn't learn much more, nor did I face any other issues during this part of the lab. I enjoyed writing this and figuring out the algorithm that will make this work. I really enjoy the different instructions built into MIPS that are all extremely useful.