

CSCE 212 Project 2 Report

Tyler Beetle

1.0 Program Input and Output

For the first program we created the required inputs from the user were the values for the first variable, a , and the second variable, b . The only output returned to the user was the values found from plugging the values into the function inside of the for loop. The second program we created we returned three total outputs and recieved zero input. The first output we put out was a string that let the user know the for loop was starting. The second output we returned let the user know the for loop was done running. The final program we created took in two inputs and returned 1 output. Similar to the first program, it took in two individual values from the user. However, the output returned was dependent on the values inputted by the user and if it passed the conditions of the if statement or not.

2.0 Program Design

The initial program we created started by asking the user to input a value for a variable. After that had been completed, the user was prompted to enter a second value for a second variable. Once both of these variables had been imputed, they were stored in temporary registers. The program then moved into a for loop that ran for a total of 5 times starting at 0. Within the for loop, the function $f = a*i + b$ was solved each time the for loop ran. The first value the user inputed was a , the second value was b , and the value of i was however times the for loop had run prior to the current time. Once the for loop had run a total of 5 times, The program moved on to print out the function to the user. The second program we created began by printing out to the user the loop had started running. A loop then ran which ran for a total of 3 times starting at 0, increasing by 2 each time and ending when it was greater than or equal to 5. Within the loop, an array was created which was set equal to the function $f = 4 - i$. The value of i

was found each time the for loop was run as the current count of the for loop times 2. After the loop has run up until greater than or equal to 5, the program prints out that the loop is finished back out to the user. The third and final program we wrote started by prompting and storing two individual values for a and b. We then utilized bge and blt to check to see if a was greater than or equal to five and b was less than 5. If both cases were true, we would move to subtract b from a. We would then print this new value, f, out to the user. However, if one or none of the cases were true, we would multiply a and b and then print this value back out to the user.

3.0 Symbol Table

| Registers | Usage |
|-----------|--|
| \$v0 | Contains address of allocated memory |
| \$a0 | The number of bytes to be allocated |
| \$s0 | Used primarily to store user input, <i>a</i> |
| \$s1 | Used primarily to store user input, <i>b</i> |
| \$s2 | Used to store user input along with output |
| \$t0 | Primarily used to store sums, products, differences, and quotients between registers |
| \$t1 | Primarily used to store sums, products, differences, and quotients between registers |
| \$t2 | Primarily used to store sums, products, differences, and quotients between registers |
| \$t3 | Primarily used to store sums, products, differences, and quotients between registers |
| \$zero | Constant value of 0 |

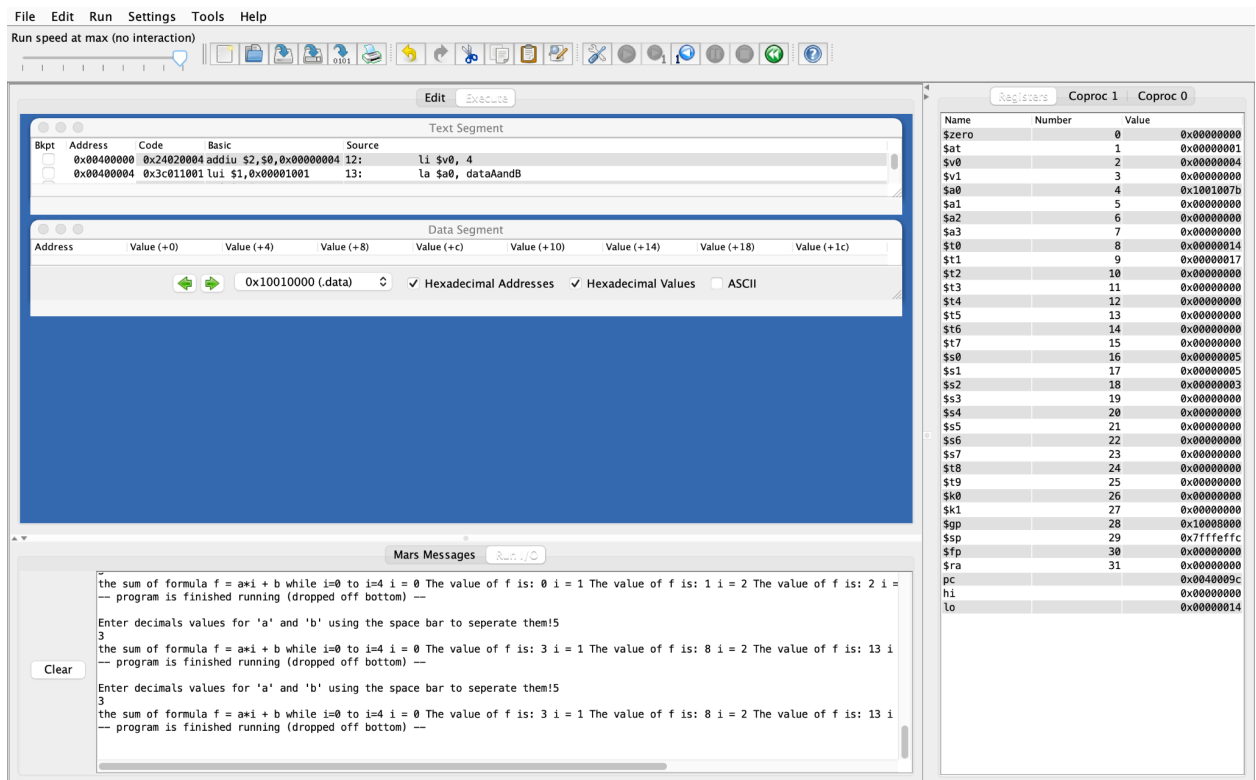
4.0 Learning Coverage

1. Learned how to utilize a loop in MIPS in order to simplify processes

2. How to create and use an array within MIPS programming
3. The strategy of creating different “methods” within MIPS in order to simplify code and to become more efficient.
4. The ability to take code from another programming language like C and be able to translate it into MIPS
5. How to simplify output by combining multiple statements into one productive statement

5.0 Test Results

5.1 Program 1 and Program 3



File Edit Run Settings Tools Help

Run speed at max (no interaction)

0101

Edit Execute

Text Segment

| Bkpt | Address | Code | Basic | Source |
|------|------------|------------|--------------------------|------------------------|
| | 0x00400000 | 0x24020004 | addiu \$2,\$0,0x00000004 | 12: li \$v0, 4 |
| | 0x00400004 | 0x3c011001 | lui \$1,0x00001001 | 13: la \$a0, dataAandB |

Data Segment

| Address | Value (+0) | Value (+4) | Value (+8) | Value (+c) | Value (+10) | Value (+14) | Value (+18) | Value (+1c) |
|--------------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|
| 0x10010000 (.data) | | | | | | | | |

Hexadecimal Addresses Hexadecimal Values ASCII

Mars Messages Run / O

Enter decimials values for 'a' and 'b' using the space bar to separte them!20
4
the sum of formula f = a*i + b while i=0 to i=4 i = 0 The value of f is: 4 i = 1 The value of f is: 24 i = 2 The value of f is: 44 i =
--- program is finished running (dropped off bottom) ---

Clear

Registers Coproc 1 Coproc 0

| Name | Number | Value |
|--------|--------|-------------|
| \$zero | 0 | 0x00000000 |
| \$at | 1 | 0x00000001 |
| \$v0 | 2 | 0x00000004 |
| \$v1 | 3 | 0x00000000 |
| \$a0 | 4 | 0x1001007b |
| \$a1 | 5 | 0x00000000 |
| \$a2 | 6 | 0x00000000 |
| \$a3 | 7 | 0x00000000 |
| \$t0 | 8 | 0x00000050 |
| \$t1 | 9 | 0x00000054 |
| \$t2 | 10 | 0x00000000 |
| \$t3 | 11 | 0x00000000 |
| \$t4 | 12 | 0x00000000 |
| \$t5 | 13 | 0x00000000 |
| \$t6 | 14 | 0x00000000 |
| \$t7 | 15 | 0x00000000 |
| \$s0 | 16 | 0x00000055 |
| \$s1 | 17 | 0x00000014 |
| \$s2 | 18 | 0x00000004 |
| \$s3 | 19 | 0x00000000 |
| \$s4 | 20 | 0x00000000 |
| \$s5 | 21 | 0x00000000 |
| \$s6 | 22 | 0x00000000 |
| \$s7 | 23 | 0x00000000 |
| \$t8 | 24 | 0x00000000 |
| \$t9 | 25 | 0x00000000 |
| \$k0 | 26 | 0x00000000 |
| \$k1 | 27 | 0x00000000 |
| \$gp | 28 | 0x10000000 |
| \$sp | 29 | 0x7ffffeffc |
| \$fp | 30 | 0x00000000 |
| \$ra | 31 | 0x00000000 |
| pc | | 0x0040005c |
| hi | | 0x00000000 |
| lo | | 0x00000050 |

File Edit Run Settings Tools Help

Run speed at max (no interaction)

0101

Edit Execute

Text Segment

| Bkpt | Address | Code | Basic | Source |
|------|------------|------------|--------------------------|-------------------|
| | 0x00400000 | 0x24020004 | addiu \$2,\$0,0x00000004 | 6: li \$v0, 4 |
| | 0x00400004 | 0x3c011001 | lui \$1,0x00001001 | 7: la \$a0, dataA |

Data Segment

| Address | Value (+0) | Value (+4) | Value (+8) | Value (+c) | Value (+10) | Value (+14) | Value (+18) | Value (+1c) |
|--------------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|
| 0x10010000 (.data) | | | | | | | | |

Hexadecimal Addresses Hexadecimal Values ASCII

Mars Messages Run / O

What is the value of a5
What is the value of b10
50
--- program is finished running (dropped off bottom) ---

Clear

Registers Coproc 1 Coproc 0

| Name | Number | Value |
|--------|--------|-------------|
| \$zero | 0 | 0x00000000 |
| \$at | 1 | 0x00000000 |
| \$v0 | 2 | 0x00000001 |
| \$v1 | 3 | 0x00000000 |
| \$a0 | 4 | 0x00000032 |
| \$a1 | 5 | 0x00000000 |
| \$a2 | 6 | 0x00000000 |
| \$a3 | 7 | 0x00000000 |
| \$t0 | 8 | 0x00000000 |
| \$t1 | 9 | 0x00000000 |
| \$t2 | 10 | 0x00000000 |
| \$t3 | 11 | 0x00000000 |
| \$t4 | 12 | 0x00000000 |
| \$t5 | 13 | 0x00000000 |
| \$t6 | 14 | 0x00000000 |
| \$t7 | 15 | 0x00000000 |
| \$s0 | 16 | 0x00000055 |
| \$s1 | 17 | 0x0000000a |
| \$s2 | 18 | 0x00000032 |
| \$s3 | 19 | 0x00000000 |
| \$s4 | 20 | 0x00000000 |
| \$s5 | 21 | 0x00000000 |
| \$s6 | 22 | 0x00000000 |
| \$s7 | 23 | 0x00000000 |
| \$t8 | 24 | 0x00000000 |
| \$t9 | 25 | 0x00000000 |
| \$k0 | 26 | 0x00000000 |
| \$k1 | 27 | 0x00000000 |
| \$gp | 28 | 0x10000000 |
| \$sp | 29 | 0x7ffffeffc |
| \$fp | 30 | 0x00000000 |
| \$ra | 31 | 0x00000000 |
| pc | | 0x00400074 |
| hi | | 0x00000000 |
| lo | | 0x00000032 |

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Text Segment

| Bkpt | Address | Code | Basic | Source |
|------|------------|------------|--------------------------|-------------------|
| | 0x00400000 | 0x24020004 | addiu \$2,\$0,0x00000004 | 6: li \$v0, 4 |
| | 0x00400004 | 0x3c011001 | lui \$1,0x00001001 | 7: la \$a0, dataA |

Data Segment

| Address | Value (+0) | Value (+4) | Value (+8) | Value (+c) | Value (+10) | Value (+14) | Value (+18) | Value (+1c) |
|--------------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|
| 0x10010000 (.data) | | | | | | | | |

Hexadecimal Addresses Hexadecimal Values ASCII

Mars Messages

Enter decimals values for 'a' and 'b' using the space bar to separate them!20
4
the sum of formula f = a*i + b while i=0 to i=4 i = 0 The value of f is: 4 i = 1 The value of f is: 24 i = 2 The value of f is: 44 i =
— program is finished running (dropped off bottom) —
What is the value of a5
What is the value of b3
2
— program is finished running (dropped off bottom) —

Registers

| Name | Number | Value |
|--------|--------|------------|
| \$zero | 0 | 0x00000000 |
| \$at | 1 | 0x00000001 |
| \$v0 | 2 | 0x00000001 |
| \$v1 | 3 | 0x00000000 |
| \$a0 | 4 | 0x00000002 |
| \$a1 | 5 | 0x00000000 |
| \$a2 | 6 | 0x00000000 |
| \$a3 | 7 | 0x00000000 |
| \$t0 | 8 | 0x00000000 |
| \$t1 | 9 | 0x00000000 |
| \$t2 | 10 | 0x00000000 |
| \$t3 | 11 | 0x00000000 |
| \$t4 | 12 | 0x00000000 |
| \$t5 | 13 | 0x00000000 |
| \$t6 | 14 | 0x00000000 |
| \$t7 | 15 | 0x00000000 |
| \$s0 | 16 | 0x00000005 |
| \$s1 | 17 | 0x00000003 |
| \$s2 | 18 | 0x00000002 |
| \$s3 | 19 | 0x00000000 |
| \$s4 | 20 | 0x00000000 |
| \$s5 | 21 | 0x00000000 |
| \$s6 | 22 | 0x00000000 |
| \$s7 | 23 | 0x00000000 |
| \$t8 | 24 | 0x00000000 |
| \$t9 | 25 | 0x00000000 |
| \$k0 | 26 | 0x00000000 |
| \$k1 | 27 | 0x00000000 |
| \$gp | 28 | 0x10000000 |
| \$sp | 29 | 0x7fffffc |
| \$fp | 30 | 0x00000000 |
| \$ra | 31 | 0x00000000 |
| pc | | 0x00000074 |
| h1 | | 0x00000000 |
| lo | | 0x00000000 |

5.2 Program 2

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Text Segment

| Bkpt | Address | Code | Basic | Source |
|------|------------|------------|--------------------------|------------------|
| | 0x00400000 | 0x24020004 | addiu \$2,\$0,0x00000004 | 8: li \$v0, 4 |
| | 0x00400004 | 0x3c011001 | lui \$1,0x00001001 | 9: la \$a0, str1 |

Data Segment

| Address | Value (+0) | Value (+4) | Value (+8) | Value (+c) | Value (+10) | Value (+14) | Value (+18) | Value (+1c) |
|--------------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|
| 0x10010000 (.data) | | | | | | | | |

Hexadecimal Addresses Hexadecimal Values ASCII

Mars Messages

What is the value of a5
What is the value of b10
50
— program is finished running (dropped off bottom) —
Loop starts!Loop ends!
— program is finished running (dropped off bottom) —

Registers

| Name | Number | Value |
|--------|--------|------------|
| \$zero | 0 | 0x00000000 |
| \$at | 1 | 0x10010000 |
| \$v0 | 2 | 0x00000004 |
| \$v1 | 3 | 0x00000000 |
| \$a0 | 4 | 0x1001000d |
| \$a1 | 5 | 0x00000000 |
| \$a2 | 6 | 0x00000000 |
| \$a3 | 7 | 0x00000000 |
| \$t0 | 8 | 0x10010018 |
| \$t1 | 9 | 0x00000000 |
| \$t2 | 10 | 0x00000000 |
| \$t3 | 11 | 0x00000004 |
| \$t4 | 12 | 0x00000000 |
| \$t5 | 13 | 0x00000000 |
| \$t6 | 14 | 0x00000000 |
| \$t7 | 15 | 0x00000000 |
| \$s0 | 16 | 0x00000006 |
| \$s1 | 17 | 0x00000000 |
| \$s2 | 18 | 0x00000000 |
| \$s3 | 19 | 0x00000000 |
| \$s4 | 20 | 0x00000000 |
| \$s5 | 21 | 0x00000000 |
| \$s6 | 22 | 0x00000000 |
| \$s7 | 23 | 0x00000000 |
| \$t8 | 24 | 0x00000000 |
| \$t9 | 25 | 0x00000000 |
| \$k0 | 26 | 0x00000000 |
| \$k1 | 27 | 0x00000000 |
| \$gp | 28 | 0x10000000 |
| \$sp | 29 | 0x7fffffc |
| \$fp | 30 | 0x00000000 |
| \$ra | 31 | 0x00000000 |
| pc | | 0x00400058 |
| h1 | | 0x00000000 |
| lo | | 0x00000000 |