Project 2 Report

Anna Phan

1.0: Program Input/Output

Program 1: The input of the program is the user putting in three positive numbers. The output would be the number that is the minimum in the list that is not zero.

Program 2: The input would the be the user's height in feet and weight in pounds. The output would be the BMI calculation based on the person's input and telling the user weather the BMI is underweight, normal weight, or overweight.

Program 3: The input is the number of homework, the average time to complete the homework in hours, number of exercises, and the average time to complete the exercise in hours. The output would be total work time in hours.

2.0: Program Design

Program 1: The program will prompt the user to enter three positive integers. The program will take three integers and compare them to one another. It will find the smallest number that is not zero and print it out for the user.

Program 2: The program will prompt the user to enter their height in feet and weight in pounds. It will take the height in feet and convert it into inches. Then the program will square the height. Multiply the weight by 703. It will convert the integer into a single precision FP so it can be calculated in the equation. the program will then divide the weight x 703 by the squared height to get the BMI. The program will determine the user's status by comparing the BMI to the appropriate BMI to get underweight, normal, or overweight.

Program 3: The program will take in the number of homework, the average time it takes to complete the homework, the number of exercises, and the average time it takes to complete the exercises. It will multiple the number of homework with the average time it takes to complete the homework. Next, it will multiple the number of exercises with the average time it takes to complete the exercises. It will then add the two numbers and tell the user the total work time.

3.0: Symbol Table

Register	Purpose & Label
$zero	To hold a constant zero
$v0	1: prints an integer 2: prints a float 4: prints a string 5: saves and integer from user
$v1	C3: total work hours from 'total' method
$a0	Argument of syscall to print string
$t0	C1: temporary place for the first number C3: temporary place for number of homeworks
$t1	C1: temporary place for the second number C3: temporary place for average time to complete homework in hours
$t2	C1: temporary place for the third number C3: temporary place for number of exercises
$t3	C3: temporary place for average time to complete exercise in hours
$s0	C2: save the weight of the user C3: saves number of homework multiplied by average time to complete homework in hours

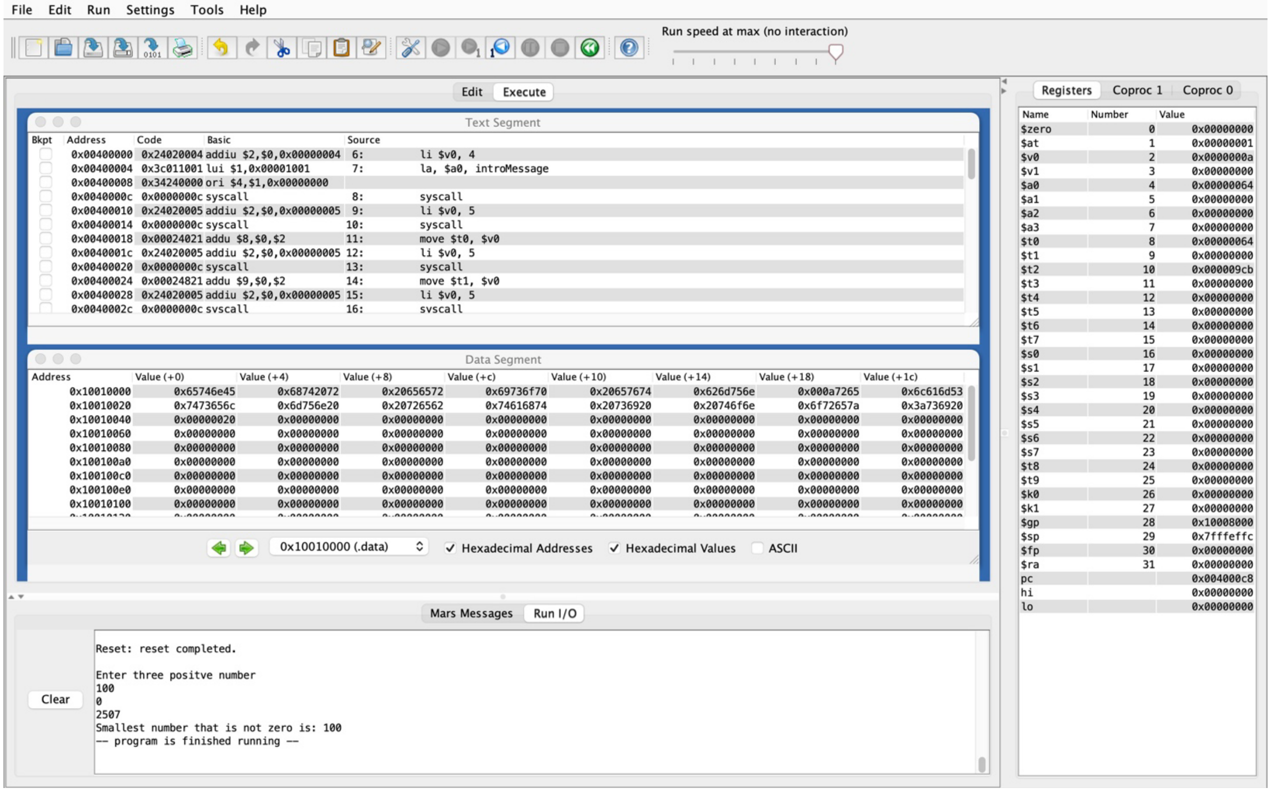
$s1	C3: saves number of exercises multiplied by average time to complete exercise in hours
$f0	C2: temporary place to pass through $f4
$f1	C2: the float for inches
$f2	C2: the float for height in feet
$f3	C2: temporary place for BMI number
$f4	C2: temporary place for height input. Temporary place for underweight number
$f5	C2: temporary place for normal wight number
$f6	C2: temporary place for overweight number
$f12	C2: passes the number so it can be printed
$f20	C2: the float for weight

4.0: Learning Coverage

1. Stacks
2. Non-leaf procedures
3. Conditional Jump Float Point
4. Data Movement Float Point
5. Understanding Branch Instructions

5.0: Test Results

Program 1:



Bkpt	Address	Code	Basic	Source	
□	0x00400000	0x24020004	addiu $2,$0,0x00000004	6:	li $v0, 4
□	0x00400004	0x3c011001	lui $1,0x00001001	7:	la, $a0, introMessage
□	0x00400008	0x34240000	ori $4,$1,0x00000000		
□	0x0040000c	0x0000000c	syscall	8:	syscall
□	0x00400010	0x24020005	addiu $2,$0,0x00000005	9:	li $v0, 5
□	0x00400014	0x0000000c	syscall	10:	syscall
□	0x00400018	0x00024021	addu $8,$0,$2	11:	move $t0, $v0
□	0x0040001c	0x24020005	addiu $2,$0,0x00000005	12:	li $v0, 5
□	0x00400020	0x0000000c	syscall	13:	syscall
□	0x00400024	0x00024821	addu $9,$0,$2	14:	move $t1, $v0
□	0x00400028	0x24020005	addiu $2,$0,0x00000005	15:	li $v0, 5
□	0x0040002c	0x0000000c	svscall	16:	svscall

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0x65746e45	0x68742072	0x20656572	0x69736f70	0x20657674	0x626d756e	0x000a7265	0x6c616d53
0x10010020	0x7473656c	0x6d756e20	0x20726562	0x74616874	0x20736920	0x20746f6e	0x6f72657a	0x3a736920
0x10010040	0x00000020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

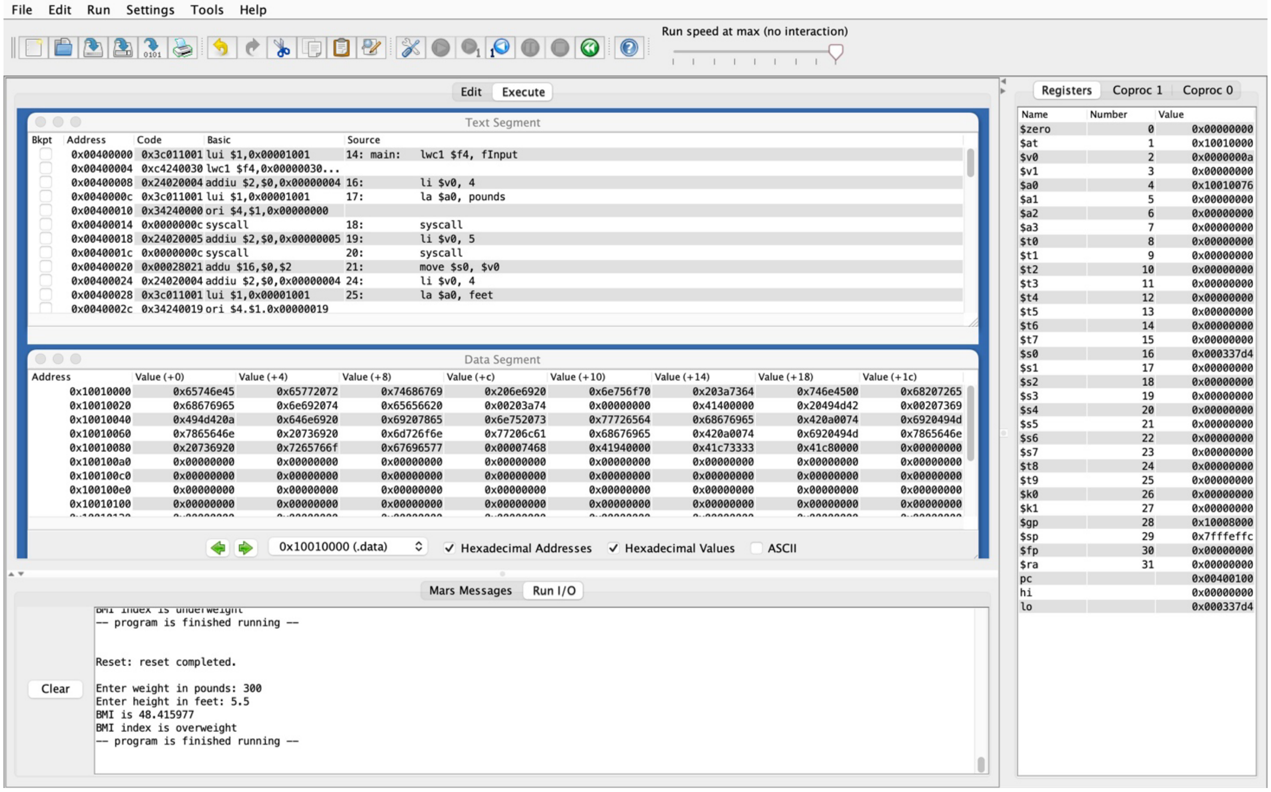
0x10010000 (.data) ☑ Hexadecimal Addresses ☑ Hexadecimal Values □ ASCII

Name	Number	Value
$zero	0	0x00000000
$at	1	0x00000001
$v0	2	0x0000000a
$v1	3	0x00000000
$a0	4	0x00000064
$a1	5	0x00000000
$a2	6	0x00000000
$a3	7	0x00000000
$t0	8	0x00000064
$t1	9	0x00000000
$t2	10	0x000009cb
$t3	11	0x00000000
$t4	12	0x00000000
$t5	13	0x00000000
$t6	14	0x00000000
$t7	15	0x00000000
$s0	16	0x00000000
$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	0x10008000
$sp	29	0x7fffeffc
$fp	30	0x00000000
$ra	31	0x00000000
pc		0x004000c8
hi		0x00000000
lo		0x00000000

```
Reset: reset completed.

Enter three positve number
100
0
2507
Smallest number that is not zero is: 100
-- program is finished running --
```

Program 2:



Bkpt	Address	Code	Basic	Source
□	0x00400000	0x3c011001	lui $1,0x00001001	14: main: lwc1 $f4, fInput
□	0x00400004	0xc4240030	lwc1 $f4,0x00000030...	
□	0x00400008	0x24020004	addiu $2,$0,0x00000004	16: li $v0, 4
□	0x0040000c	0x3c011001	lui $1,0x00001001	17: la $a0, pounds
□	0x00400010	0x34240000	ori $4,$1,0x00000000	
□	0x00400014	0x0000000c	syscall	18: syscall
□	0x00400018	0x24020005	addiu $2,$0,0x00000005	19: li $v0, 5
□	0x0040001c	0x0000000c	syscall	20: syscall
□	0x00400020	0x00028021	addu $16,$0,$2	21: move $s0, $v0
□	0x00400024	0x24020004	addiu $2,$0,0x00000004	24: li $v0, 4
□	0x00400028	0x3c011001	lui $1,0x00001001	25: la $a0, feet
□	0x0040002c	0x34240019	ori $4.$1.0x00000019	

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0x65746e45	0x65772072	0x74686769	0x206e6920	0x6e756f70	0x203a7364	0x746e4500	0x68207265
0x10010020	0x68676965	0x6e692074	0x65656620	0x00203a74	0x00000000	0x41400000	0x20494d42	0x00207369
0x10010040	0x494d420a	0x646e6920	0x69207865	0x6e752073	0x77726564	0x68676965	0x420a0074	0x6920494d
0x10010060	0x7865646e	0x20736920	0x6d726f6e	0x77206c61	0x68676965	0x420a0074	0x6920494d	0x7865646e
0x10010080	0x20736920	0x7265766f	0x67696577	0x00007468	0x41940000	0x41c73333	0x41c80000	0x00000000
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

0x10010000 (.data)   ✓ Hexadecimal Addresses   ✓ Hexadecimal Values   □ ASCII

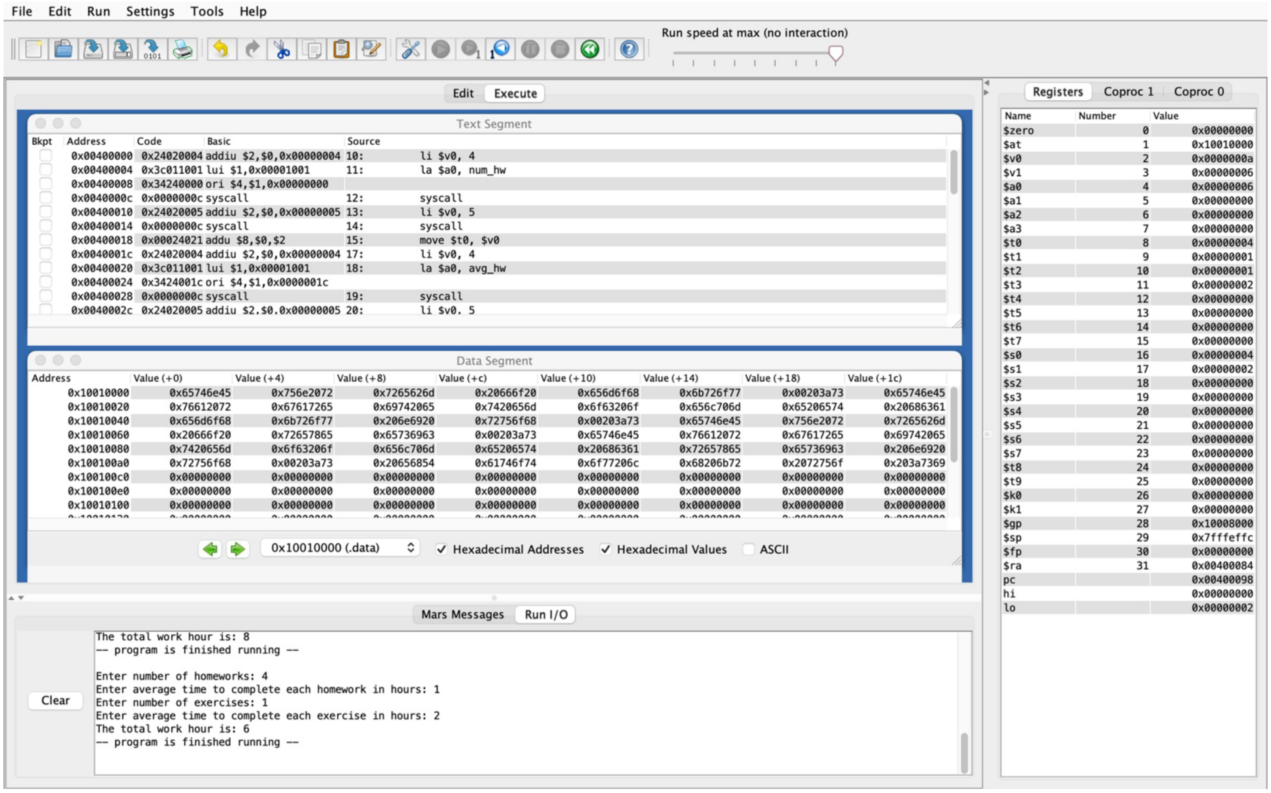
```
BMI index is underweight
-- program is finished running --


Reset: reset completed.

Enter weight in pounds: 300
Enter height in feet: 5.5
BMI is 48.415977
BMI index is overweight
-- program is finished running --
```

Name	Number	Value
$zero	0	0x00000000
$at	1	0x10010000
$v0	2	0x0000000a
$v1	3	0x00000000
$a0	4	0x10010076
$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	0x000337d4
$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	0x10008000
$sp	29	0x7fffeffc
$fp	30	0x00000000
$ra	31	0x00000000
pc		0x00400100
hi		0x00000000
lo		0x000337d4

Program 3:



Bkpt	Address	Code	Basic	Source
□	0x00400000	0x24020004	addiu $2,$0,0x00000004	10: li $v0, 4
□	0x00400004	0x3c011001	lui $1,0x00001001	11: la $a0, num_hw
□	0x00400008	0x34240000	ori $4,$1,0x00000000	
□	0x0040000c	0x0000000c	syscall	12: syscall
□	0x00400010	0x24020005	addiu $2,$0,0x00000005	13: li $v0, 5
□	0x00400014	0x0000000c	syscall	14: syscall
□	0x00400018	0x00024021	addu $8,$0,$2	15: move $t0, $v0
□	0x0040001c	0x24020004	addiu $2,$0,0x00000004	17: li $v0, 4
□	0x00400020	0x3c011001	lui $1,0x00001001	18: la $a0, avg_hw
□	0x00400024	0x3424001c	ori $4,$1,0x0000001c	
□	0x00400028	0x0000000c	syscall	19: syscall
□	0x0040002c	0x24020005	addiu $2.$0.0x00000005	20: li $v0. 5

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0x65746e45	0x756e2072	0x7265626d	0x20666f20	0x656d6f68	0x6b726f77	0x00203a73	0x65746e45
0x10010020	0x76612072	0x67617265	0x69742065	0x7420656d	0x6f63206f	0x656c706d	0x65206574	0x20686361
0x10010040	0x656d6f68	0x6b726f77	0x206e6920	0x72756f68	0x00203a73	0x65746e45	0x756e2072	0x7265626d
0x10010060	0x20666f20	0x72657865	0x65736963	0x00203a73	0x65746e45	0x76612072	0x67617265	0x69742065
0x10010080	0x7420656d	0x6f63206f	0x656c706d	0x65206574	0x20686361	0x72657865	0x65736963	0x206e6920
0x100100a0	0x72756f68	0x00203a73	0x20656854	0x61746f74	0x6f77206c	0x68206b72	0x2072756f	0x203a7369
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

0x10010000 (.data)   ✓ Hexadecimal Addresses   ✓ Hexadecimal Values   □ ASCII

```
The total work hour is: 8
-- program is finished running --

Enter number of homeworks: 4
Enter average time to complete each homework in hours: 1
Enter number of exercises: 1
Enter average time to complete each exercise in hours: 2
The total work hour is: 6
-- program is finished running --
```

Name	Number	Value
$zero	0	0x00000000
$at	1	0x10010000
$v0	2	0x0000000a
$v1	3	0x00000006
$a0	4	0x00000006
$a1	5	0x00000000
$a2	6	0x00000000
$a3	7	0x00000000
$t0	8	0x00000004
$t1	9	0x00000001
$t2	10	0x00000001
$t3	11	0x00000002
$t4	12	0x00000000
$t5	13	0x00000000
$t6	14	0x00000000
$t7	15	0x00000000
$s0	16	0x00000004
$s1	17	0x00000002
$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	0x10008000
$sp	29	0x7fffeffc
$fp	30	0x00000000
$ra	31	0x00400084
pc		0x00400098
hi		0x00000000
lo		0x00000002