

# CCCN 221 – Computer Architecture

## LAB#6 Task5

Task Date: As per BB

Submission Date: As per BB

Student Name: Amin Yahya Selhabi

Student ID: 2140632

**Note: Student must attach the code and the screenshot of the Final output using MIPS or Qtscmpm. Copy and cheating will be mark "0"**

Marks:

Exercises	1	2	3	Total
Allocated	1	1	1	3
Obtained				
CLO, PLO, SO	3.1, V3, S05	3.1, V3, S05	3.1, V3, S05	

### 1. Task

Write a program that prompts the user to enter an integer and determines whether it is divisible by 5 and 6 but not both. For example, if your input is 10, the output should be. *Read Lab6 sheet.*

(Hint: Use the AND , OR ,X\_OR operators)

**screenshot of the Final output using MIPS or Qtscmpm (without screenshot marks will be deducted).**

Output Sample.

```
Is 10 divisible by 5 and 6? false
Is 10 divisible by 5 or 6? true
Is 10 divisible by 5 or 6, but not both? True
```

### 2. Task

Implement the following assembly program in Mars MIPS that performs the following tasks:

**Write a program that prints the numbers from 1 to given number N. Read Lab sheets.**

**screenshot of the Final output using MIPS or Qtscmpm (without screenshot marks will be deducted).**

3. Implement the following assembly program in Mars MIPS that performs the following tasks:

**Write a program that asks user to enter Numbers and displays the Average. Read Lab sheets.**

**screenshot of the Final output using MIPS or Qtscmpm (without screenshot marks will be deducted)**

## Contents

Task 1: .....	3
Task 2: .....	5
Task 3: .....	6
Task 1 code: .....	7
Task 2 code: .....	12
Task 3 code: .....	13
References .....	15

## Code included in the end

### Task 1:

**Task1.asm**

```
1 .data
2 msg: .ascii "Please Enter a Number: "
3 is: .ascii "is "
4 andd: .ascii "divisible by 5 and 6?"
5 orr: .ascii "divisible by 5 or 6?"
6 xorrr: .ascii "divisible by 5 or 6, but not both?"
7 true: .ascii "True"
8 false: .ascii "False"
9
10
11 .text
12 # Asking for input
13 li t0, 4
14 la t40, msg
15 syscall
16 li t0, 5
17 syscall
18 move t40, t0
19
20 # Dividing by 5
21 div t0, t40, 5
22 mfti t41
23
24 # Checking if remainder is bigger than zero
25 bnez t41, True1
26 move t41, t40 # zero for false
27 j Skip1
28
29 True1:
30 addi t41, t41, 1 # i for true
31
32 Skip1:
33 # Dividing by 6
34 li t0, 1
35 div t0, t40, 6
36 mfti t42
37
38 # Logic check
39 and t43, t41, t42
40 orr t43, t41, t42
41 xorrr t43, t41, t42
42
43 # Print results
44 li t0, 4
45 la t40, is
46 move t40, t43
47 syscall
48 li t0, 4
49 la t40, andd
50 move t40, t43
51 syscall
52 li t0, 4
53 la t40, xorrr
54 move t40, t43
55 syscall
56
57 # Print "True" or "False"
58 bnez t43, True2
59 move t41, t40 # zero for false
60 j Skip2
61
62 True2:
63 addi t41, t41, 1 # i for true
64
65 Skip2:
66
67
68 .end
69
```

**Registers**

Register	Number	Value
\$zero	0	0x00000000
\$at	1	0x00100000
\$v0	2	0x00000004
\$v1	3	0x00000000
\$a0	4	0x00100072
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000005
\$t1	9	0x00000000
\$t2	10	0x00000000
\$t3	11	0x00100018
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$t8	16	0x00000004
\$t9	17	0x00000000
\$s0	18	0x00000000
\$s1	19	0x00000000
\$s2	20	0x00000000
\$s3	21	0x00000000
\$s4	22	0x00000000
\$s5	23	0x00000000
\$s6	24	0x00000000
\$s7	25	0x00000000
\$s8	26	0x00000000
\$s9	27	0x00000000
\$fp	28	0x00000000
\$gp	29	0x00000000
\$tp	30	0x00000000
\$k0	31	0x00000000
\$hi		0x00400194
\$lo		0x00000004

**Mars Messages**

```
Please Enter a Number: 10
Is 10 divisible by 5 and 6? False
Is 10 divisible by 5 or 6? True
Is 10 divisible by 5 or 6, but not both? True
-- program is finished running --
```

**Task1.asm**

```
30 li t0, 1
31 div t0, t40, 5
32 mfti t41
33
34 # Checking if remainder is bigger than zero
35 bnez t41, True1
36 move t41, t40 # zero for false
37 j End1
38
39 True1:
40 addi t41, t41, 1 # i for true
41
42 End1:
43
44 # LOGIC Check if both t41, t42 are 1 print true for AND
45 and t43, t41, t42
46 orr t43, t41, t42
47 xorrr t43, t41, t42
48
49 # Print results
50 li t0, 4
51 la t40, is
52 move t40, t43
53 syscall
54 li t0, 4
55 la t40, andd
56 move t40, t43
57 syscall
58 li t0, 4
59 la t40, xorrr
60 move t40, t43
61 syscall
62
63 # Print "True" or "False"
64 bnez t43, True2
65 move t41, t40 # zero for false
66 j Skip2
67
68 True2:
69 addi t41, t41, 1 # i for true
70
71 Skip2:
72
73
74 .end
75
```

**Registers**

Register	Number	Value
\$zero	0	0x00000000
\$at	1	0x00100000
\$v0	2	0x00000004
\$v1	3	0x00000000
\$a0	4	0x00100072
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000005
\$t1	9	0x00000000
\$t2	10	0x00000000
\$t3	11	0x00100018
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$t8	16	0x00000004
\$t9	17	0x00000000
\$s0	18	0x00000000
\$s1	19	0x00000000
\$s2	20	0x00000000
\$s3	21	0x00000000
\$s4	22	0x00000000
\$s5	23	0x00000000
\$s6	24	0x00000000
\$s7	25	0x00000000
\$s8	26	0x00000000
\$s9	27	0x00000000
\$fp	28	0x00000000
\$gp	29	0x00000000
\$tp	30	0x00000000
\$k0	31	0x00000000
\$hi		0x00400194
\$lo		0x00000004

**Mars Messages**

```
Reset: reset completed.
Please Enter a Number: 30
Is 30 divisible by 5 and 6? True
Is 30 divisible by 5 or 6? True
Is 30 divisible by 5 or 6, but not both? False
-- program is finished running (dropped off bottom) --
```

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Task3.asm Task1.asm

```

67
68 andZero: # Prints "True"
69 li t0, 4
70 la t40, true
71 syscall
72 j ocCheck
73
74
75 andOne: # Prints "False"
76 li t0, 4
77 la t40, false
78 syscall
79 j ocCheck # Rabbit
80
81
82
83 ocCheck:
84 # Prints out "Is Num divisible by 5 and 6?"
85 li t0, 4
86 la t13, is
87 move t40, t13
88 syscall
89 li t0, 1
90 move t40, t40
91 syscall
92 li t0, 4
93 la t40, not
94 syscall
95
96 # LOGIC Check if both t41, t42 are 1 print true for OR
97 beqz t41, orZero
98 beqz t42, orZero
99 j orOne
100
101 orZero:
102 li t0, 4
103
104
Line: 1 Column: 1 Show Line Numbers

```

Mars Messages Run IO

Reset: reset completed.

Please Enter a Number: 30

Clear

Is 30 divisible by 5 and 6? True  
 Is 30 divisible by 5 or 6? True  
 Is 30 divisible by 5 or 6, but not both? False  
 -- program is finished running (dropped off bottom) --

Registers	Coproc 1	Coproc 0	Name	Number	Value
\$zero			0	0	0x00000000
\$at			1	1	0x00100000
\$v0			2	2	0x00000004
\$v1			3	3	0x00000000
\$a0			4	4	0x10010078
\$a1			5	5	0x00000000
\$a2			6	6	0x00000000
\$a3			7	7	0x00000000
\$t0			8	8	0x00000000
\$t1			9	9	0x00000000
\$t2			10	10	0x00000000
\$t3			11	11	0x10010018
\$t4			12	12	0x00000000
\$t5			13	13	0x00000000
\$t6			14	14	0x00000000
\$t7			15	15	0x00000000
\$t8			16	16	0x00000014
\$t9			17	17	0x00000000
\$s0			18	18	0x00000000
\$s1			19	19	0x00000000
\$s2			20	20	0x00000000
\$s3			21	21	0x00000000
\$s4			22	22	0x00000000
\$s5			23	23	0x00000000
\$s6			24	24	0x00000000
\$s7			25	25	0x00000000
\$s8			26	26	0x00000000
\$s9			27	27	0x00000000
\$fp			28	28	0x10000000
\$gp			29	29	0x7ffffc00
\$tp			30	30	0x00000000
\$ra			31	31	0x00000000
\$lo					0x00000014
\$hi					0x00000000
\$lo					0x00000005

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Task3.asm Task1.asm

```

90 beqz t41, orZero
91 j orOne
100
101 orZero:
102 li t0, 4
103 la t40, true
104 syscall
105 j xocCheck
106
107 orOne:
108 li t0, 4
109 la t40, false
110 syscall
111 j xocCheck # Rabbit
112
113 xocCheck:
114 # Prints out "Is Num divisible by 5 and 6 but not both?"
115 li t0, 4
116 la t13, is
117 move t40, t13
118 syscall
119 li t0, 1
120 move t40, t40
121 syscall
122 li t0, 4
123 la t40, not
124 syscall
125
126 # LOGIC Check if both t41, t42 are 1 print true for XOR
127 beqz t41, xocZero
128 beqz t42, xocZero
129 j xocFalse
130
131 xocZero:
132 beqz t40, xocFalse
133 li t0, 4
134
Line: 1 Column: 1 Show Line Numbers

```

Mars Messages Run IO

Reset: reset completed.

Please Enter a Number: 30

Clear

Is 30 divisible by 5 and 6? True  
 Is 30 divisible by 5 or 6? True  
 Is 30 divisible by 5 or 6, but not both? False  
 -- program is finished running (dropped off bottom) --

Registers	Coproc 1	Coproc 0	Name	Number	Value
\$zero			0	0	0x00000000
\$at			1	1	0x00100000
\$v0			2	2	0x00000004
\$v1			3	3	0x00000000
\$a0			4	4	0x10010078
\$a1			5	5	0x00000000
\$a2			6	6	0x00000000
\$a3			7	7	0x00000000
\$t0			8	8	0x00000000
\$t1			9	9	0x00000000
\$t2			10	10	0x00000000
\$t3			11	11	0x10010018
\$t4			12	12	0x00000000
\$t5			13	13	0x00000000
\$t6			14	14	0x00000000
\$t7			15	15	0x00000000
\$t8			16	16	0x00000014
\$t9			17	17	0x00000000
\$s0			18	18	0x00000000
\$s1			19	19	0x00000000
\$s2			20	20	0x00000000
\$s3			21	21	0x00000000
\$s4			22	22	0x00000000
\$s5			23	23	0x00000000
\$s6			24	24	0x00000000
\$s7			25	25	0x00000000
\$s8			26	26	0x00000000
\$s9			27	27	0x00000000
\$fp			28	28	0x10000000
\$gp			29	29	0x7ffffc00
\$tp			30	30	0x00000000
\$ra			31	31	0x00000000
\$lo					0x00000014
\$hi					0x00000000
\$lo					0x00000005

Task1.asm

```

110 syscall
111 xorCheck #Habit
112
113 xorCheck:
114 # Print out "Is num divisible by 5 and 6 but not both?"
115 li t0, 4
116 la t3, is
117 move t40, t3
118 syscall
119 li t0, 1
120 move t40, t40
121 syscall
122 li t0, 4
123 la t40, not
124 syscall
125
126 # LOGIC Check if both t41, t42 are 1 print true for XOR
127 beq t41, notZero
128 beq t42, notZero
129 xorFalse
130
131 xorZero:
132 beq t40, notFalse
133 li t0, 4
134 la t40, true
135 syscall
136 li t0, 10
137 syscall
138
139 xorFalse:
140 li t0, 4
141 la t40, false
142 syscall
143 # Awesome homework!
144

```

Registers

Register	Number	Value
\$zero	0	0x00000000
\$at	1	0x00000000
\$v0	2	0x00000004
\$v1	3	0x00000000
\$a0	4	0x00000000
\$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
\$f0	16	0x00000000
\$f1	17	0x00000000
\$f2	18	0x00000000
\$f3	19	0x00000000
\$f4	20	0x00000000
\$f5	21	0x00000000
\$f6	22	0x00000000
\$f7	23	0x00000000
\$f8	24	0x00000000
\$f9	25	0x00000000
\$f10	26	0x00000000
\$f11	27	0x00000000
\$f12	28	0x00000000
\$f13	29	0x00000000
\$f14	30	0x00000000
\$f15	31	0x00000000
\$PC		0x00400000
\$HI		0x00000000
\$LO		0x00000000

Mars Messages

```

Reset: reset completed.
Please Enter a Number: 30
Is 30 divisible by 5 and 6? True
Is 30 divisible by 5 or 6? True
Is 30 divisible by 5 or 6, but not both? False
-- program is finished running (dropped off bottom) --

```

## Task 2:

Task2.asm

```

1 # This program will count from 0 to the given positive integer
2 .data
3 num: .asciiz "Please enter a number: "
4 count: .word -1
5 space: .asciiz " "
6
7 # Print "Please enter a number: "
8 .text
9 li v0, 4
10 la t0, num
11 syscall
12 # Input
13 li v0, 5
14 syscall
15 move t40, v0
16
17 # Printing numbers from 0 to given positive integer
18 li t0, count
19 la t1, space
20
21 loop:
22 addi t0, t0, 1
23 li v0, 1
24 move t40, t0
25 syscall
26 # Printing Blank " "
27 li t0, 4
28 move t40, t1
29 syscall
30 bit t0, t40, loop #looping
31
32 li v0, 10
33 syscall
34

```

Registers

Register	Number	Value
\$zero	0	0x00000000
\$at	1	0x00000000
\$v0	2	0x00000004
\$v1	3	0x00000000
\$a0	4	0x00000000
\$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
\$f0	16	0x00000000
\$f1	17	0x00000000
\$f2	18	0x00000000
\$f3	19	0x00000000
\$f4	20	0x00000000
\$f5	21	0x00000000
\$f6	22	0x00000000
\$f7	23	0x00000000
\$f8	24	0x00000000
\$f9	25	0x00000000
\$f10	26	0x00000000
\$f11	27	0x00000000
\$f12	28	0x00000000
\$f13	29	0x00000000
\$f14	30	0x00000000
\$f15	31	0x00000000
\$PC		0x00400000
\$HI		0x00000000
\$LO		0x00000000

Mars Messages

```

Please enter a number: 9
0 1 2 3 4 5 6 7 8 9
-- program is finished running --

```

## Task 3:

C:\Users\ahmed\OneDrive\Desktop\Computer Architecture LAB\LAB06\Task3.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Task2.asm Task3.asm

```

1 # This program will calculate the average of giving numbers
2 .data
3 max: .word 10 "Please enter a number to continue or a character to quit: "
4 average: .word 0 "The average is: "
5 remainder: .word 0 "The remainder is: "
6 error: .word 0
7
8 # Prints "Please enter a number: "
9 .text
10 .loop
11 lw $t3, error # To stop the loop when an error is caught
12 Loop:
13 li $v0, 4
14 la $a0, max
15 syscall
16
17 li $v0, 5
18 syscall
19 move $t0, $v0
20 add $t0, $t0, $t0 # Sum of inputs
21 add $t1, $t1, 1 # Counting the inputs
22 j Loop
23
24 # Special thanks to https://courses.missouristate.edu/jmvolz/mars/help/MarsExceptions.html for helping me figuring out how to catch an error
25 # $tval 0x00001100 # Default location of Exception Handler
26
27 # Prints "The average is: "
28 li $v0, 4
29 la $a0, average
30 syscall
31 # Calculates and prints the quotient
32 li $v0, 1
33 div $t0, $t1
34 mflo $a0
35 syscall
36
37 # Prints "The remainder is: "
38 li $v0, 4
39 la $a0, remainder
40 syscall
41
42 li $v0, 1
43 mfhi $a0
44 syscall
45 li $v0, 10
46 syscall
47
48 # Coprocessor 0 register $t14 has address of trapping instruction
49 mfcr $t0, $t14
50 # Add 4 to point to next instruction
51 addi $t0, $t0, 4
52 # Store new address back into $t14
53 mfcr $t0, $t14
54 # Error return: set PC to value in $t14
55 j $t0
56 .exit
57
58 .msg: .ascii "Flag generated"
59
60

```

Line: 28 Column: 30 Show Line Numbers

Mars Messages Run IO

Please enter a number to continue or a character to quit: 10  
Please enter a number to continue or a character to quit: 20  
Please enter a number to continue or a character to quit: 30  
Please enter a number to continue or a character to quit: a  
Clear  
The average is: 20  
The remainder is: 0  
-- program is finished running --

Registers	Coproc 1	Coproc 0	Name	Number	Value
\$zero			0	0	0x00000000
\$at			1	1	0x00100000
\$v0			2	2	0x0000000a
\$v1			3	3	0x00000000
\$a0			4	4	0x00000000
\$a1			5	5	0x00000000
\$a2			6	6	0x00000000
\$a3			7	7	0x00000000
\$t0			8	8	0x0000003c
\$t1			9	9	0x00000013
\$t2			10	10	0x00000000
\$t3			11	11	0x00000000
\$t4			12	12	0x00000000
\$t5			13	13	0x00000000
\$t6			14	14	0x00000000
\$t7			15	15	0x00000000
\$t8			16	16	0x0000001a
\$t9			17	17	0x00000000
\$s0			18	18	0x00000000
\$s1			19	19	0x00000000
\$s2			20	20	0x00000000
\$s3			21	21	0x00000000
\$s4			22	22	0x00000000
\$s5			23	23	0x00000000
\$s6			24	24	0x00000000
\$s7			25	25	0x00000000
\$s8			26	26	0x00000000
\$s9			27	27	0x00000000
\$fp			28	28	0x00000000
\$gp			29	29	0x7ffffc00
\$tp			30	30	0x00000000
\$t8			31	31	0x00000000
\$t9			32	32	0x00000000
\$t0			33	33	0x00000000
\$t1			34	34	0x00000000
\$t2			35	35	0x00000000
\$t3			36	36	0x00000000
\$t4			37	37	0x00000000
\$t5			38	38	0x00000000
\$t6			39	39	0x00000000
\$t7			40	40	0x00000000
\$t8			41	41	0x00000000
\$t9			42	42	0x00000000
\$t0			43	43	0x00000000
\$t1			44	44	0x00000000
\$t2			45	45	0x00000000
\$t3			46	46	0x00000000
\$t4			47	47	0x00000000
\$t5			48	48	0x00000000
\$t6			49	49	0x00000000
\$t7			50	50	0x00000000
\$t8			51	51	0x00000000
\$t9			52	52	0x00000000
\$t0			53	53	0x00000000
\$t1			54	54	0x00000000
\$t2			55	55	0x00000000
\$t3			56	56	0x00000000
\$t4			57	57	0x00000000
\$t5			58	58	0x00000000
\$t6			59	59	0x00000000
\$t7			60	60	0x00000000
\$t8			61	61	0x00000000
\$t9			62	62	0x00000000
\$t0			63	63	0x00000000
\$t1			64	64	0x00000000
\$t2			65	65	0x00000000
\$t3			66	66	0x00000000
\$t4			67	67	0x00000000
\$t5			68	68	0x00000000
\$t6			69	69	0x00000000
\$t7			70	70	0x00000000
\$t8			71	71	0x00000000
\$t9			72	72	0x00000000
\$t0			73	73	0x00000000
\$t1			74	74	0x00000000
\$t2			75	75	0x00000000
\$t3			76	76	0x00000000
\$t4			77	77	0x00000000
\$t5			78	78	0x00000000
\$t6			79	79	0x00000000
\$t7			80	80	0x00000000
\$t8			81	81	0x00000000
\$t9			82	82	0x00000000
\$t0			83	83	0x00000000
\$t1			84	84	0x00000000
\$t2			85	85	0x00000000
\$t3			86	86	0x00000000
\$t4			87	87	0x00000000
\$t5			88	88	0x00000000
\$t6			89	89	0x00000000
\$t7			90	90	0x00000000
\$t8			91	91	0x00000000
\$t9			92	92	0x00000000
\$t0			93	93	0x00000000
\$t1			94	94	0x00000000
\$t2			95	95	0x00000000
\$t3			96	96	0x00000000
\$t4			97	97	0x00000000
\$t5			98	98	0x00000000
\$t6			99	99	0x00000000
\$t7			100	100	0x00000000
\$t8			101	101	0x00000000
\$t9			102	102	0x00000000
\$t0			103	103	0x00000000
\$t1			104	104	0x00000000
\$t2			105	105	0x00000000
\$t3			106	106	0x00000000
\$t4			107	107	0x00000000
\$t5			108	108	0x00000000
\$t6			109	109	0x00000000
\$t7			110	110	0x00000000
\$t8			111	111	0x00000000
\$t9			112	112	0x00000000
\$t0			113	113	0x00000000
\$t1			114	114	0x00000000
\$t2			115	115	0x00000000
\$t3			116	116	0x00000000
\$t4			117	117	0x00000000
\$t5			118	118	0x00000000
\$t6			119	119	0x00000000
\$t7			120	120	0x00000000
\$t8			121	121	0x00000000
\$t9			122	122	0x00000000
\$t0			123	123	0x00000000
\$t1			124	124	0x00000000
\$t2			125	125	0x00000000
\$t3			126	126	0x00000000
\$t4			127	127	0x00000000
\$t5			128	128	0x00000000
\$t6			129	129	0x00000000
\$t7			130	130	0x00000000
\$t8			131	131	0x00000000
\$t9			132	132	0x00000000
\$t0			133	133	0x00000000
\$t1			134	134	0x00000000
\$t2			135	135	0x00000000
\$t3			136	136	0x00000000
\$t4			137	137	0x00000000
\$t5			138	138	0x00000000
\$t6			139	139	0x00000000
\$t7			140	140	0x00000000
\$t8			141	141	0x00000000
\$t9			142	142	0x00000000
\$t0			143	143	0x00000000
\$t1			144	144	0x00000000
\$t2			145	145	0x00000000
\$t3			146	146	0x00000000
\$t4			147	147	0x00000000
\$t5			148	148	0x00000000
\$t6			149	149	0x00000000
\$t7			150	150	0x00000000
\$t8			151	151	0x00000000
\$t9			152	152	0x00000000
\$t0			153	153	0x00000000
\$t1			154	154	0x00000000
\$t2			155	155	0x00000000
\$t3			156	156	0x00000000
\$t4			157	157	0x00000000
\$t5			158	158	0x00000000
\$t6			159	159	0x00000000
\$t7			160	160	0x00000000
\$t8			161	161	0x00000000
\$t9			162	162	0x00000000
\$t0			163	163	0x00000000
\$t1			164	164	0x00000000
\$t2			165	165	0x00000000
\$t3			166	166	0x00000000
\$t4			167	167	0x00000000
\$t5			168	168	0x00000000
\$t6			169	169	0x00000000
\$t7			170	170	0x00000000
\$t8			171	171	0x00000000
\$t9			172	172	0x00000000
\$t0			173	173	0x00000000
\$t1			174	174	0x00000000
\$t2			175	175	0x00000000
\$t3			176	176	0x00000000
\$t4			177	177	0x00000000
\$t5			178	178	0x00000000
\$t6			179	179	0x00000000
\$t7			180	180	0x00000000
\$t8			181	181	0x00000000
\$t9			182	182	0x00000000
\$t0			183	183	0x00000000
\$t1			184	184	0x00000000
\$t2			185	185	0x00000000
\$t3			186	186	0x00000000
\$t4			187	187	0x00000000
\$t5			188	188	0x00000000
\$t6			189	189	0x00000000
\$t7			190	190	0x00000000
\$t8			191	191	0x00000000
\$t9			192	192	0x00000000
\$t0			193	193	0x00000000
\$t1			194	194	0x00000000
\$t2			195	195	0x00000000
\$t3			196	196	0x00000000
\$t4			197	197	0x00000000
\$t5			198	198	0x00000000
\$t6			199	199	0x00000000
\$t7			200	200	0x00000000
\$t8			201	201	0x00000000
\$t9			202	202	0x00000000
\$t0			203	203	0x00000000
\$t1			204	204	0x00000000
\$t2			205	205	0x00000000
\$t3			206	206	0x00000000
\$t4			207	207	0x00000000
\$t5			208	208	0x00000000
\$t6			209	209	0x00000000
\$t7			210	210	0x00000000
\$t8			211	211	0x00000000
\$t9			212	212	0x00000000
\$t0			213	213	0x00000000
\$t1			214	214	0x00000000
\$t2			215	215	0x00000000
\$t3			216	216	0x00000000
\$t4			217	217	0x00000000
\$t5			218	218	0x00000000
\$t6			219	219	0x00000000
\$t7			220	220	0x00000000
\$t8			221	221	0x00000000
\$t9			222	222	0x00000000
\$t0			223	223	0x00000000
\$t1			224	224	0x00000000
\$t2			225	225	0x00000000
\$t3			226	226	0x00000000
\$t4			227	227	0x00000000
\$t5			228	228	0x00000000
\$t6					

## Task 1 code:

.data

Num: .asciiz "Please Enter a Number: "

is: .asciiz "\nIs "

andd: .asciiz " divisible by 5 and 6? "

orr: .asciiz " divisible by 5 or 6? "

xorr: .asciiz " divisible by 5 or 6, but not both?? "

true: .asciiz "True "

false: .asciiz "False "

.text

# Asking for input

li \$v0, 4

la \$a0, Num

syscall

li \$v0, 5

syscall

move \$s0, \$v0

# Dividing by 5

div \$t0, \$s0, 5

mfhi \$s1

# Checking if reminder is bigger than zero

bnez \$s1, True1

move \$t1, \$zero # zero for false

j Skip1

True1:

addi \$s1, \$s1, 1 # 1 for true

Skip1:

# Dividing by 6

li \$v0, 1

div \$t0, \$s0, 6

mfhi \$s2

# Checking if reminder is bigger than zero

bnez \$s2, True2

move \$t2, \$zero # zero for false

j End

True2:

addi \$s2, \$s2, 1 # 1 for true

j End # Habbit

# LOGIC Check if both \$s1, \$s2 are 1 print true for AND

End:

or \$t0, \$s1, \$s2

# Prints out "Is Num divisible by 5 and 6?"

li \$v0, 4

la \$t3, is

move \$a0, \$t3

syscall

li \$v0, 1



```
move $a0, $s0
```

```
syscall
```

```
li $v0, 4
```

```
la $a0, andd
```

```
syscall
```

```
# Checking if 5 and 6 are divisible
```

```
beqz $t0, andZero
```

```
j andOne
```

```
andZero: # Prints "True"
```

```
li $v0, 4
```

```
la $a0, true
```

```
syscall
```

```
j orCheck
```

```
andOne: # Prints "False"
```

```
li $v0, 4
```

```
la $a0, false
```

```
syscall
```

```
j orCheck # Habbit
```

```
orCheck:
```

```
# Prints out "Is Num divisible by 5 and 6?"
```

```
li $v0, 4
```

```
la $t3, is
```

```
move $a0, $t3
```

```
syscall
```

```
li $v0, 1
```

```
move $a0, $s0
```

```
syscall
```

```
li $v0, 4
```

```
la $a0, orr
```

```
syscall
```

```
# LOGIC Check if both $s1, $s2 are 1 print true for OR
```

```
beqz $s1, orZero
```

```
beqz $s2, orZero
```

```
j orOne
```

```
orZero:
```

```
li $v0, 4
```

```
la $a0, true
```

```
syscall
```

```
j xorCheck
```

```
orOne:
```

```
li $v0, 4
```

```
la $a0, false
```

```
syscall
```

```
j xorCheck # Habbit
```

```
xorCheck:
```

```
# Prints out "Is Num divisible by 5 and 6 but not both??"
```

```
li $v0, 4
```

```

la $t3, is
move $a0, $t3
syscall
li $v0, 1
move $a0, $s0
syscall
li $v0, 4
la $a0, xor
syscall

# LOGIC Check if both $s1, $s2 are 1 print true for XOR
beqz $s1, xorZero
beqz $s2, xorZero
j xorFalse

xorZero:
beqz $t0, xorFalse
li $v0, 4
la $a0, true
syscall
li $v0, 10
syscall

xorFalse:
li $v0, 4
la $a0, false
syscall

# Awesome homework!

```

## Task 2 code:

# This programme will count from 0 to the giving positive integer

.data

num: .asciiz "Please enter a number: "

count: .word -1

space: .asciiz " "

# Print "Please enter a number: "

.text

li \$v0, 4

la \$a0, num

syscall

# Input

li \$v0, 5

syscall

move \$s0, \$v0

# Printing numbers from 0 to giving positive integer

lw \$t0, count

la \$t1, space

loop:

addi \$t0, \$t0, 1

li \$v0, 1

move \$a0, \$t0

syscall

# Printing Blank " "

li \$v0, 4

move \$a0, \$t1

```
syscall
```

```
blt $t0, $s0, loop #looping
```

```
li $v0, 10
```

```
syscall
```

### Task 3 code:

```
# This programme will calculate the average of giving numbers
```

```
.data
```

```
num: .asciiz "Please enter a number to continue or a character to quit: "
```

```
average: .asciiz "\nThe average is: "
```

```
remainder: .asciiz "\nThe remainder is: "
```

```
error: .word 0
```

```
# Prints "Please enter a number: "
```

```
.text
```

```
lw $t3, error # To stop the loop when an error is caught
```

```
Loop:
```

```
li $v0, 4
```

```
la $a0, num
```

```
syscall
```

```
li $v0, 5
```

```
syscall
```

```
move $s0, $v0
```

```
add $t0, $t0, $s0 # Sum of inputs
```

```
add $t1, $t1, 1 # Counting the inputs
```

```
j Loop
```

# Special thanks to <https://courses.missouristate.edu/kenvollmar/mars/help/MarsExceptions.html> for helping me figuring out how to catch an error

.ktext 0x80000180 # Default Location of Exception Handler

# Prints "\nThe average is: "

li \$v0, 4

la \$a0, average

syscall

# Calculates and prints the quotient

li \$v0, 1

div \$t0, \$t1

mflo \$a0

syscall

# Prints "\nThe remainder is: "

li \$v0, 4

la \$a0, remainder

syscall

li \$v0, 1

mfhi \$a0

syscall

li \$v0, 10

syscall

# Coprocessor 0 register \$14 has address of trapping instruction

mfc0 \$k0,\$14

# Add 4 to point to next instruction

addi \$k0,\$k0, 4

```
# Store new address back into $14
mtc0 $k0,$14

# Error return; set PC to value in $14
eret

.kdata
msg: .asciiz "Trap generated\n"
```

## References

University, M. S. (n.d.). Retrieved from  
<https://courses.missouristate.edu/kenvollmar/mars/help/MarsExceptions.html>