

COS-10004 (Computer Systems)

Name: Dave Nguyen (Nguyen Quang Anh).

ID: 104697710.

Q8.1.1:

Program

```
1| MOV R0, #15
2| STR R0, .WriteUnsignedNum
3| MOV R0, #msg1
4| STR R0, .WriteString
5| HALT
6| msg1: .ASCIIZ "remaining\n"
```

Processor

PC 20
LR 0
SP 1048576
R12 0
R11 0
R10 0
R9 0
R8 0
R7 0
R6 0
R5 0
R4 0
R3 0
R2 0
R1 0
R0 20

▶ || ◻

⏩ ▶ ⚙

Count 5

Current Instruction

Status bits NZCV 0000

Input/Output

15 remaining

Program HALTED. STOP, LOAD or EDIT

Q8.1.2-3:

Program

```
1| MOV R0, #15
2| loop: STR R0, .WriteUnsignedNum
3| MOV R1, #t1
4| STR R1, .WriteString
5| MOV R1, #t2
6| STR R1, .WriteString
7| LDR R2, .InputNum
8| SUB R0, R0, R2
9| B loop
10| HALT
11| t1: .ASCIIZ "remaining\n"
12| t2: .ASCIIZ "How many do you want to remove(1-3)?\n"
```

Processor

PC 0x00000000
LR 0x00000000
SP 0x00100000
R12 0x00000000
R11 0x00000000
R10 0x00000000
R9 0x00000000
R8 0x00000000
R7 0x00000000
R6 0x00000000
R5 0x00000000
R4 0x00000000
R3 0x00000000
R2 0x00000000
R1 0x00000000
R0 0x00000000

▶ || ◻

⏩ ▶

Count 0

Current Instruction

Status bits NZCV 0000

Input/Output

Stop done, edit & Submit, RUN/STEP or alter memory

Memory

000	0x0	0x4	0x8	0xc
0x0000	0xe3a0000f	0xe50f00f8	0xe3a01028	0xe50f10f4
0x0001	0xe3a01033	0xe50f10fc	0xe51f2118	0xe0400002
0x0002	0xeaafffff7	0xe1000070	0xe616d6572	0xe6e696e69
0x0003	0x48000a67	0x5d20776f	0x20796e61	0x79206f64
0x0004	0x7220756f	0x2074615e	0x72206f74	0x766f6d65
0x0005	0x2d312865	0x0a3f2933	0x00000000	0x00000000
0x0006	0x00000000	0x00000000	0x00000000	0x00000000
0x0007	0x00000000	0x00000000	0x00000000	0x00000000
0x0008	0x00000000	0x00000000	0x00000000	0x00000000
0x0009	0x00000000	0x00000000	0x00000000	0x00000000
0x000a	0x00000000	0x00000000	0x00000000	0x00000000
0x000b	0x00000000	0x00000000	0x00000000	0x00000000
0x000c	0x00000000	0x00000000	0x00000000	0x00000000
0x000d	0x00000000	0x00000000	0x00000000	0x00000000
0x000e	0x00000000	0x00000000	0x00000000	0x00000000
0x000f	0x00000000	0x00000000	0x00000000	0x00000000
0x0010	0x00000000	0x00000000	0x00000000	0x00000000
0x0011	0x00000000	0x00000000	0x00000000	0x00000000
0x0012	0x00000000	0x00000000	0x00000000	0x00000000
0x0013	0x00000000	0x00000000	0x00000000	0x00000000
0x0014	0x00000000	0x00000000	0x00000000	0x00000000
0x0015	0x00000000	0x00000000	0x00000000	0x00000000
0x0016	0x00000000	0x00000000	0x00000000	0x00000000
0x0017	0x00000000	0x00000000	0x00000000	0x00000000
0x0018	0x00000000	0x00000000	0x00000000	0x00000000
0x0019	0x00000000	0x00000000	0x00000000	0x00000000
0x001a	0x00000000	0x00000000	0x00000000	0x00000000
0x001b	0x00000000	0x00000000	0x00000000	0x00000000
0x001c	0x00000000	0x00000000	0x00000000	0x00000000
0x001d	0x00000000	0x00000000	0x00000000	0x00000000
0x001e	0x00000000	0x00000000	0x00000000	0x00000000
0x001f	0x00000000	0x00000000	0x00000000	0x00000000

Binary Clear

Q8.2.1:

When the number of matchsticks is a negative one ($-x$, $x > 0$), the output will be $(2^{32} - x)$

The screenshot displays a MIPS simulator interface with three main sections: Program, Processor, and Input/Output.

Program Section: Contains assembly code for a loop that prints the number of remaining matchsticks and prompts the user for removals.

```
1 | MOV R0, #15
2 | loop: STR R0, .WriteUnsignedNum
3 | MOV R1, #t1
4 | STR R1, .WriteString
5 | MOV R1, #t2
6 | STR R1, .WriteString
7 | int: LDR R2, .InputNum
8 | CMP R2, #1
9 | BLT int
10 | CMP R2, #3
11 | BGT int
12 | SUB R0, R0, R2
13 | B loop
14 | HALT
15 | t1: .ASCIIZ "remaining\n"
16 | t2: .ASCIIZ "How many do you want to remove(1-3)?"
```

Processor Section: Shows the state of the MIPS processor. The PC is 0x0000001c. Registers R0-R15 are listed with their values. The Count register is 7. The Current Instruction is empty. The Status bits are NZCV 0000.

Input/Output Section: Displays the output of the program, showing "15 remaining" and the prompt "How many do you want to remove(1-3)?". The input field is empty, with the text "Input expected" below it.

Q8.2.2:

- If the input is equal to 0 because of $R2 = 0 < 1$, then the N status bit will be 1.
- Two assembly instructions can be used to create a branch that only occurs under these conditions: BGT: Z clear, N and V the same. BLT: N and V different.
- If the first condition is not ($R2 > 0$), and R2 is negative then $N = 1$.
If the first condition is not ($R2 > 0$), and $R2 = 0$ then $Z = 1$.
If the second condition is not ($R2 < 4$), and $R2 > 4$ then $C = 1$.
If the second condition is not ($R2 < 4$), and $R2 = 4$ then both $Z \& C = 1$.
-

Program

```

1  MOV R0, #15
2  STR R0, .WriteUnsignedNum
3  MOV R1, #msg1
4  STR R1, .WriteString
5  Loop:
6  MOV R1, #msg2
7  STR R1, .WriteString
8  LDR R2, .InputNum
9  start:
10  CMP R2, #0
11  BGT else1      // if R2 > 0 then jump to label else1
12  B invalid1
13  else1:
14  CMP R2, #4
15  BLT cont      // if R2 < 4 then jump to label cont
16  B invalid1
17  invalid1:
18  MOV R1, #msg3
19  STR R1, .WriteString
20  LDR R2, .InputNum
21  B start
22  cont:
23  SUB R0, R0, R2
24  STR R0, .WriteUnsignedNum
25  MOV R1, #msg1
26  STR R1, .WriteString
27  B Loop
28  HALT
29  msg1: .ASCIZ "remaining\n"
30  msg2: .ASCIZ "How many do you want to remove (1-3)?\n"
31  msg3: .ASCIZ "Please input a valid number!\n"

```

Processor

PC 0x0000001c
 LR 0x00000000
 SP 0x00100000
 R12 0x00000000
 R11 0x00000000
 R10 0x00000000
 R9 0x00000000
 R8 0x00000000
 R7 0x00000000
 R6 0x00000000
 R5 0x00000000
 R4 0x00000000
 R3 0x00000000
 R2 0x00000000
 R1 0x00000067
 R0 0x0000000f



Count
 Current Instruction
 Status bits NZCV 0000

Input/Output

15 remaining
 How many do you want to remove (1-3)?

Program

```

1  MOV R0, #15
2  STR R0, .WriteUnsignedNum
3  MOV R1, #msg1
4  STR R1, .WriteString
5  Loop:
6  MOV R1, #msg2
7  STR R1, .WriteString
8  LDR R2, .InputNum
9  start:
10  CMP R2, #0
11  BGT else1      // if R2 > 0 then jump to label else1
12  B invalid1
13  else1:
14  CMP R2, #4
15  BLT cont      // if R2 < 4 then jump to label cont
16  B invalid1
17  invalid1:
18  MOV R1, #msg3
19  STR R1, .WriteString
20  LDR R2, .InputNum
21  B start
22  cont:
23  SUB R0, R0, R2
24  STR R0, .WriteUnsignedNum
25  MOV R1, #msg1
26  STR R1, .WriteString
27  B Loop
28  HALT
29  msg1: .ASCIZ "remaining\n"
30  msg2: .ASCIZ "How many do you want to remove (1-3)?\n"
31  msg3: .ASCIZ "Please input a valid number!\n"

```

Processor

PC 0x0000001c
 LR 0x00000000
 SP 0x00100000
 R12 0x00000000
 R11 0x00000000
 R10 0x00000000
 R9 0x00000000
 R8 0x00000000
 R7 0x00000000
 R6 0x00000000
 R5 0x00000000
 R4 0x00000000
 R3 0x00000000
 R2 0x00000003
 R1 0x00000067
 R0 0x0000000c



Count
 Current Instruction
 Status bits NZCV 1000

Input/Output

How many do you want to remove (1-3)?
 12 remaining
 How many do you want to remove (1-3)?




Q8.3.1:




Program

```
1 select:
2   LDR R4, .Random
3   LSL R4, R4, #30
4   LSR R4, R4, #30
5   CMP R4, #0
6   BGT conti
7   B select
8 conti:
9   STR R4, .WriteUnsignedNum
10  HALT
```

Processor

PC	0x00000020
LR	0x00000000
SP	0x00100000
R12	0x00000000
R11	0x00000000
R10	0x00000000
R9	0x00000000
R8	0x00000000
R7	0x00000000
R6	0x00000000
R5	0x00000000
R4	0x00000001
R3	0x00000000
R2	0x00000000
R1	0x00000000
R0	0x00000000





Count

13

Current Instruction

Status bits

NZCV
0000

Input/Output

1
Program HALTED. STOP, LOAD or EDIT




Q8.3.2:




Program

```
1   MOV R0, #3
2 select:
3   LDR R4, .Random
4   LSL R4, R4, #30
5   LSR R4, R4, #30
6   CMP R4, #0
7   BGT conti
8   B select
9 conti:
10  CMP R4, R0
11  BGT select
12  B continue
13 continue:
14  STR R4, .WriteUnsignedNum
15  HALT
```

Processor

PC	0x00000030
LR	0x00000000
SP	0x00100000
R12	0x00000000
R11	0x00000000
R10	0x00000000
R9	0x00000000
R8	0x00000000
R7	0x00000000
R6	0x00000000
R5	0x00000000
R4	0x00000002
R3	0x00000000
R2	0x00000000
R1	0x00000000
R0	0x00000003





Count

11

Current Instruction

Status bits

NZCV
1000

Input/Output

2
Program HALTED. STOP, LOAD or EDIT