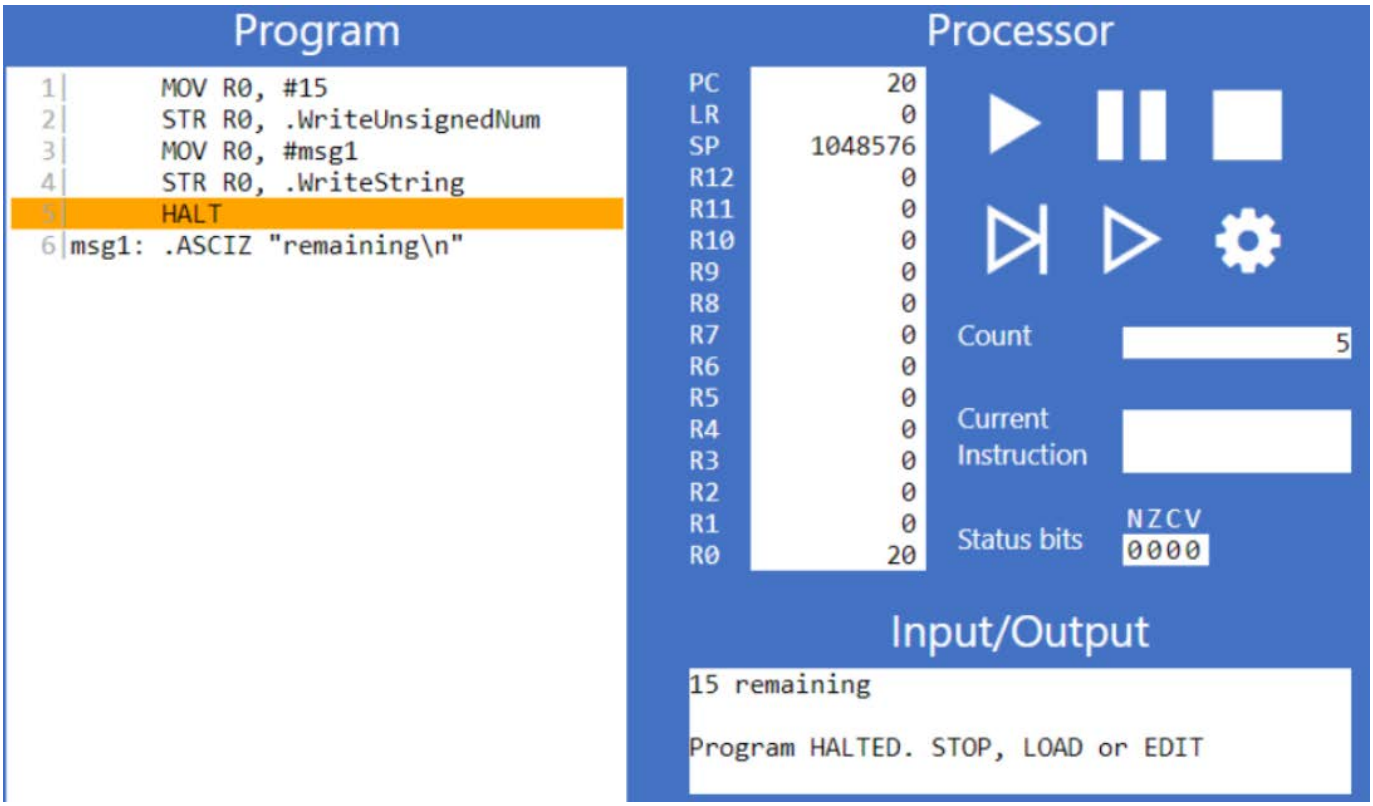
COS-10004 (Computer Systems)

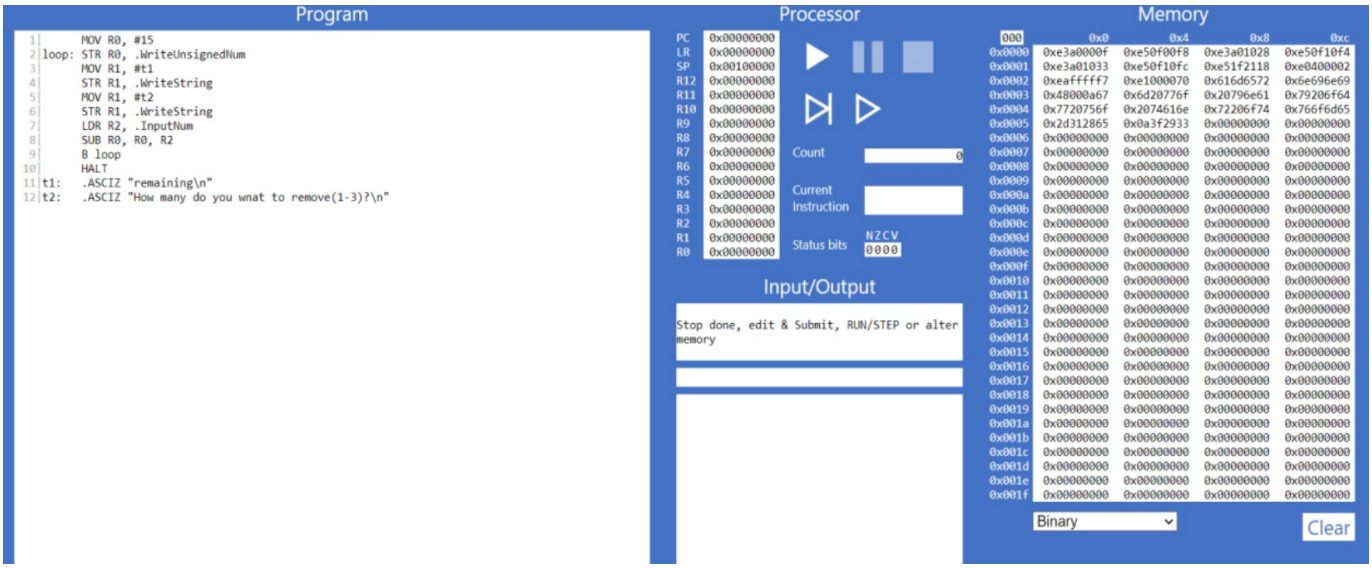
Name: Dave Nguyen (Nguyen Quang Anh).

ID: 104697710.

Q8.1.1:

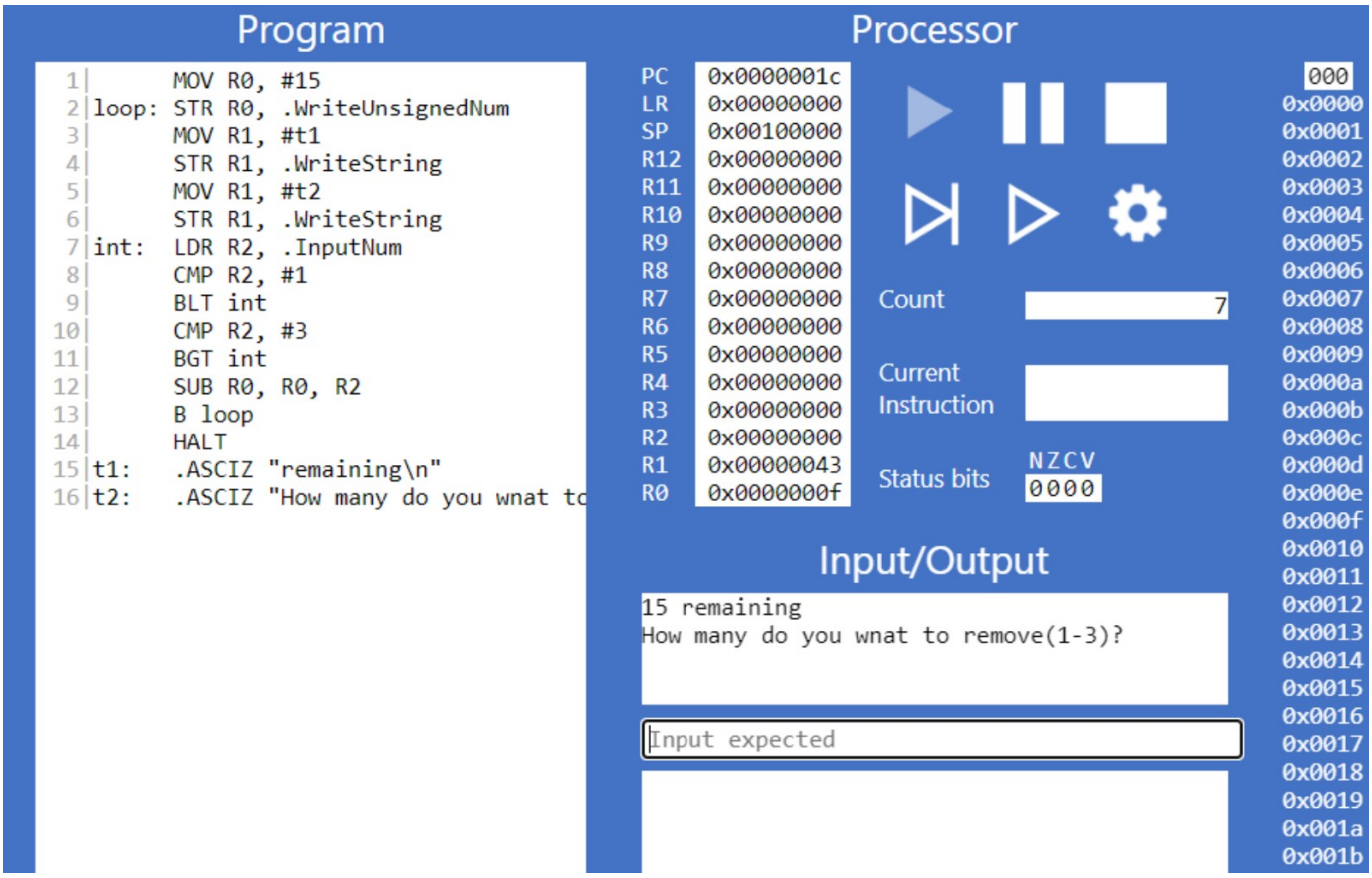


Q8.1.2-3:



Q8.2.1:

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



Q8.2.2:

a. If the input is equal to 0 because of R2 = 0 < 1, then the N status bit will be 1.
b. 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.
c. 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.
d.

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: 7

Current Instruction:

Status bits: NZCV 0000

Input/Output

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

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: 19

Current Instruction:

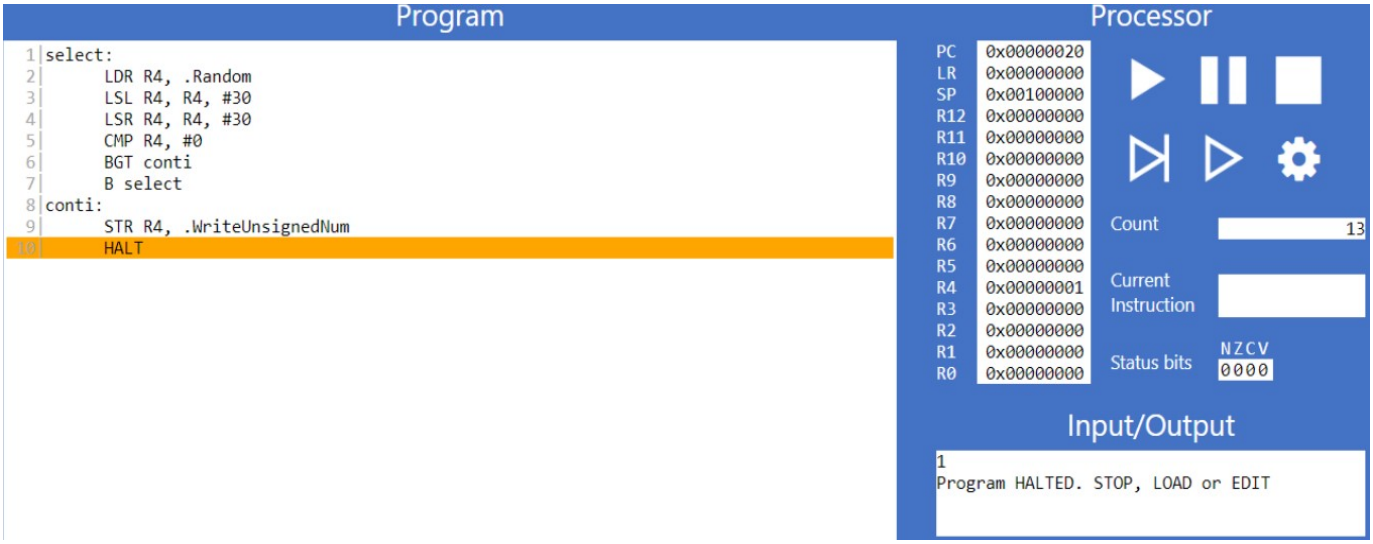
Status bits: NZCV 1000

Input/Output

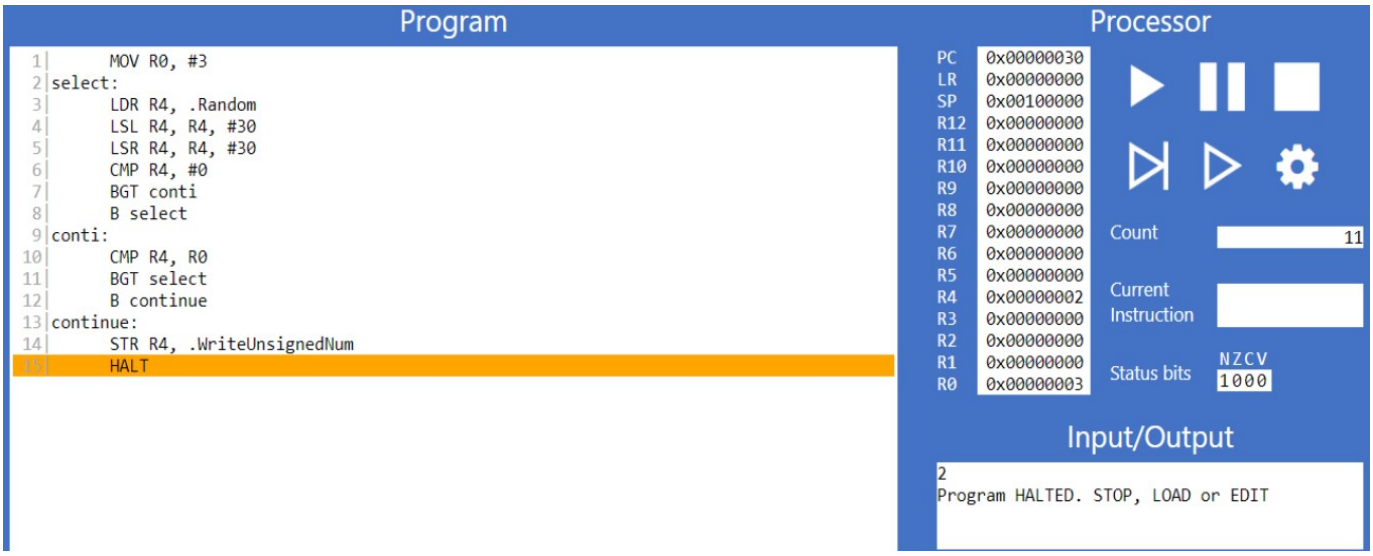
```
How many do you want to remove (1-3)?
12 remaining
How many do you want to remove (1-3)?
```

Input expected

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