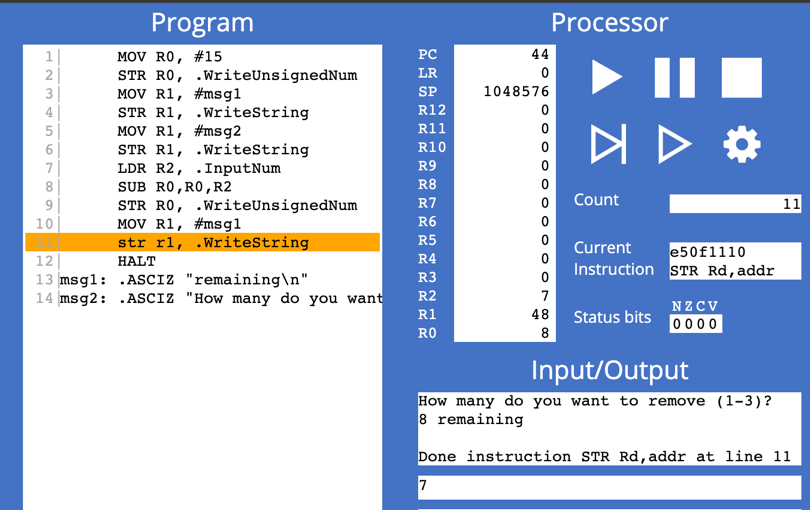
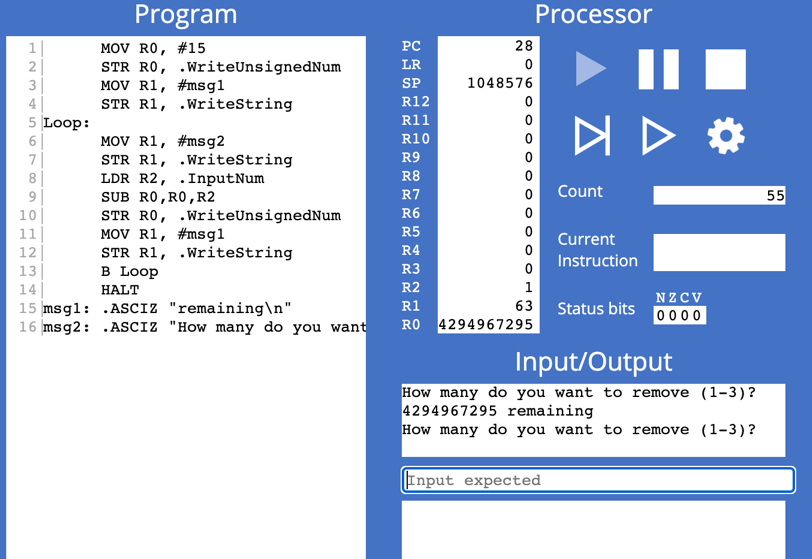
**Exercise 8.1.1 – 8.1.3**

****

**Exercise 8.2.1**

****

If we enter a number beyond 0, let’s say we want to remove 1 when there are 0 matchsticks left. It stills works normally as the actual value in register shows -1, meanwhile the register shows a big number in unsigned decimal.

**Exercise 8.2.2**

1. 0 < R2 < 4
2. Two assembly instructions that can be used are: BGT and BLT

BGT: Z clear, N and V are the same.

BLT: N and V different

1. If the first condition is not met (R2 > 0), and R2 is negative, then N = 1. If the first condition is not met (R2 > 0), and R2 = 0 then Z = 1

If the second condition is not met (R2 < 4), and R2 > 4 then C = 1. If the second condition is not met (R2 < 4), and R2 = 4 then both Z = 1 and C = 1.

1. 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 jump to else1

12| B invalid1

13|else1:

14| CMP R2, #4

15| BLT cont //If R2 < 4 jump to 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

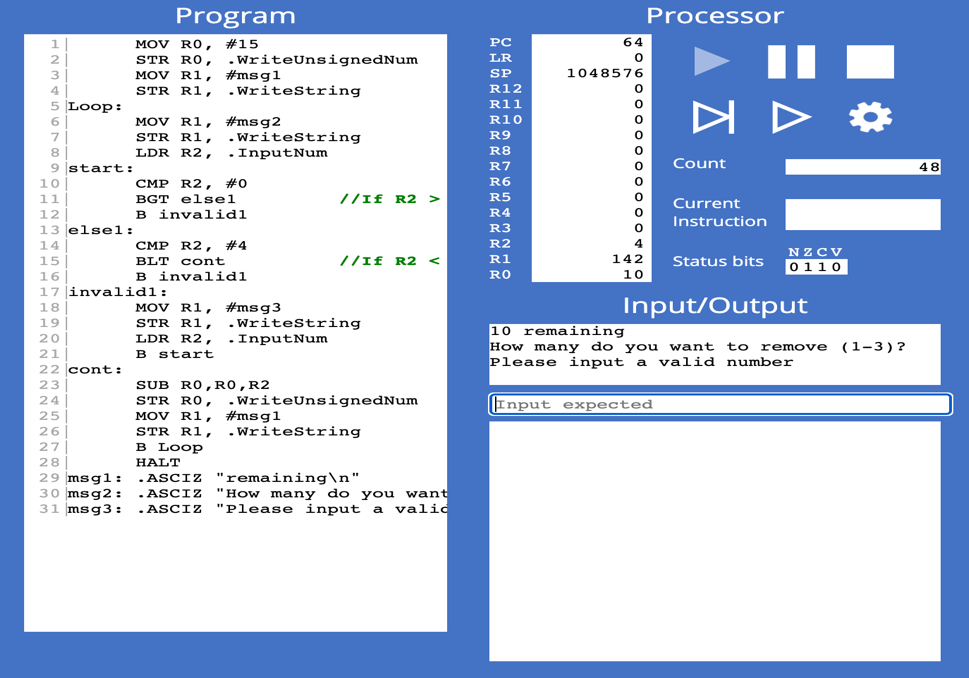
27| B Loop

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"



**Exercise 8.3.1**

1. LSL R4, R4, #30

LSR R4,R4, #30

1. Select:

LDR R4, .Random

LSL R4,R4, #30

LSL R4,R4, #30

CMP R4, #0

BGT conti

B select

Conti:

STR R4, .WriteUnsignedNum

HALTGraphical user interface, application, Word

Description automatically generated

**Exercise 8.3.2**

MOV R0, #3

Select:

LDR R4, .Random

LSL R4, R4, #30

LSR R4,R4, #30

CMP R4, #0

BGT conti

B select

Conti:

CMP R4, R0

BGT select

B continue

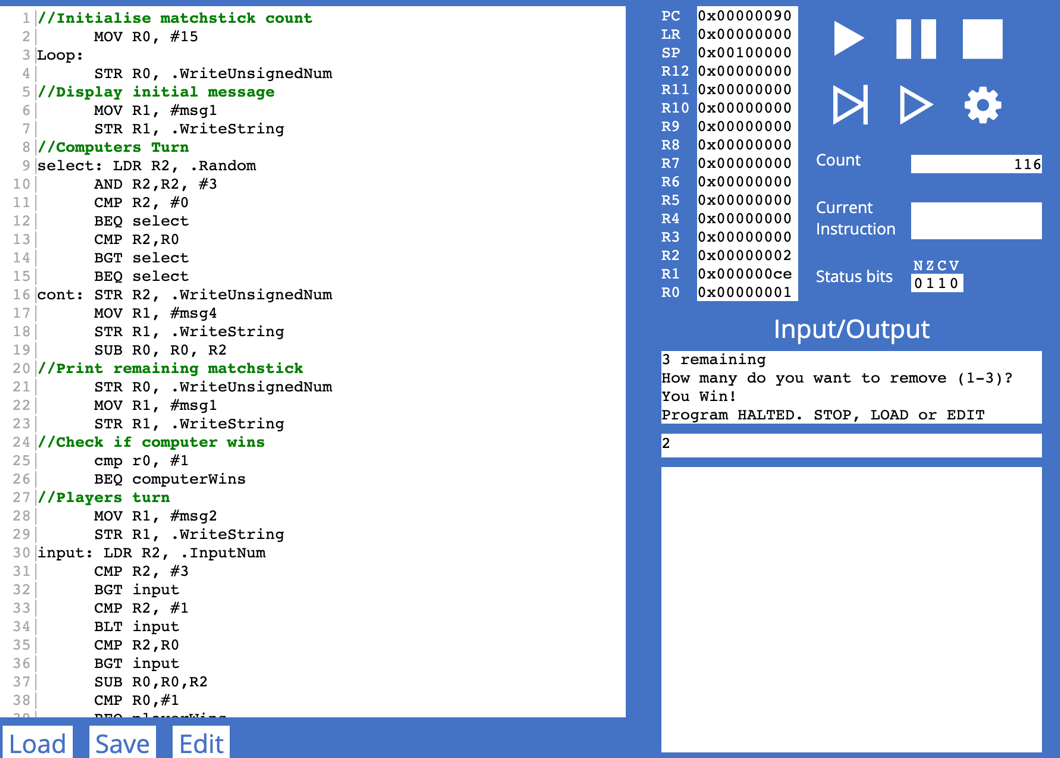
Continue:

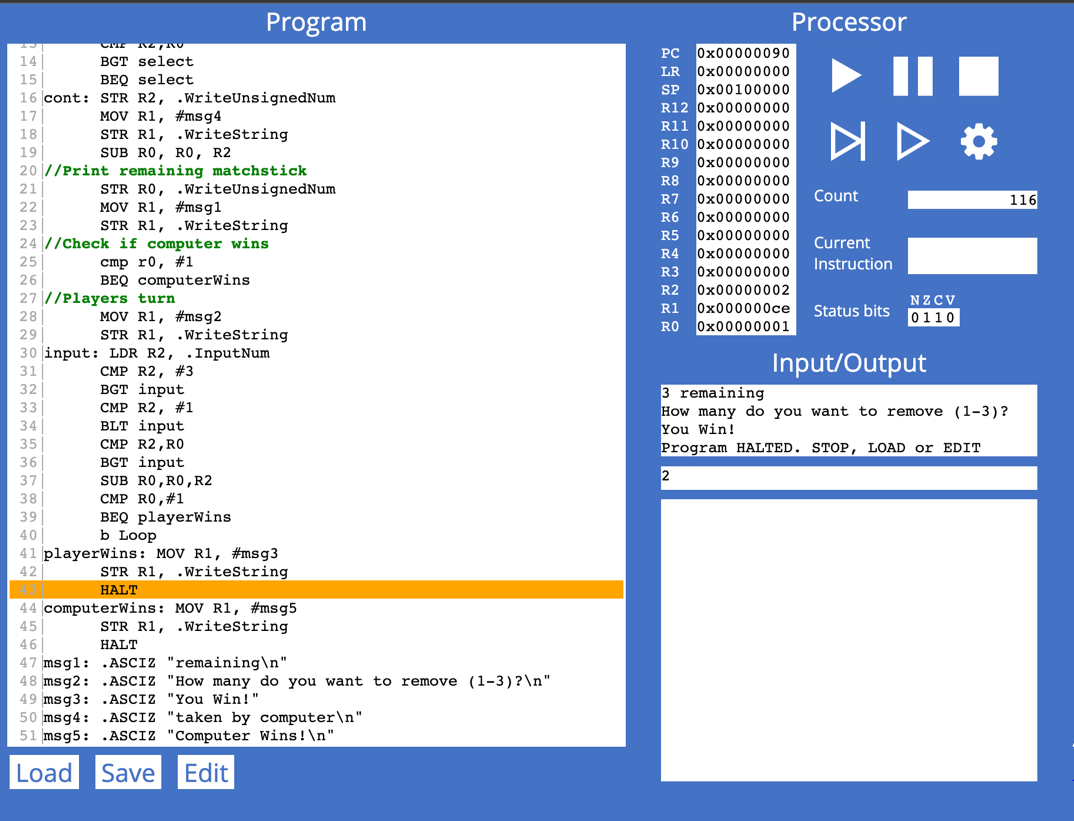
STR R4, .WriteUnsignedNum

HALTGraphical user interface, application

Description automatically generated

**Exercise 8.4.1 – 8.4.2**

****

****

1|//Initialise matchstick count

2| MOV R0, #15

3|Loop:

4| STR R0, .WriteUnsignedNum

5|//Display initial message

6| MOV R1, #msg1

7| STR R1, .WriteString

8|//Computers Turn

9|select: LDR R2, .Random

10| AND R2,R2, #3

11| CMP R2, #0

12| BEQ select

13| CMP R2,R0

14| BGT select

15| BEQ select

16|cont: STR R2, .WriteUnsignedNum

17| MOV R1, #msg4

18| STR R1, .WriteString

19| SUB R0, R0, R2

20|//Print remaining matchstick

21| STR R0, .WriteUnsignedNum

22| MOV R1, #msg1

23| STR R1, .WriteString

24|//Check if computer wins

25| cmp r0, #1

26| BEQ computerWins

27|//Players turn

28| MOV R1, #msg2

29| STR R1, .WriteString

30|input: LDR R2, .InputNum

31| CMP R2, #3

32| BGT input

33| CMP R2, #1

34| BLT input

35| CMP R2,R0

36| BGT input

37| SUB R0,R0,R2

38| CMP R0,#1

39| BEQ playerWins

40| b Loop

41|playerWins: MOV R1, #msg3

42| STR R1, .WriteString

43| HALT

44|computerWins: MOV R1, #msg5

45| STR R1, .WriteString

46| HALT

47|msg1: .ASCIZ "remaining\n"

48|msg2: .ASCIZ "How many do you want to remove (1-3)?\n"

49|msg3: .ASCIZ "You Win!"

50|msg4: .ASCIZ "taken by computer\n"

51|msg5: .ASCIZ "Computer Wins!\n"