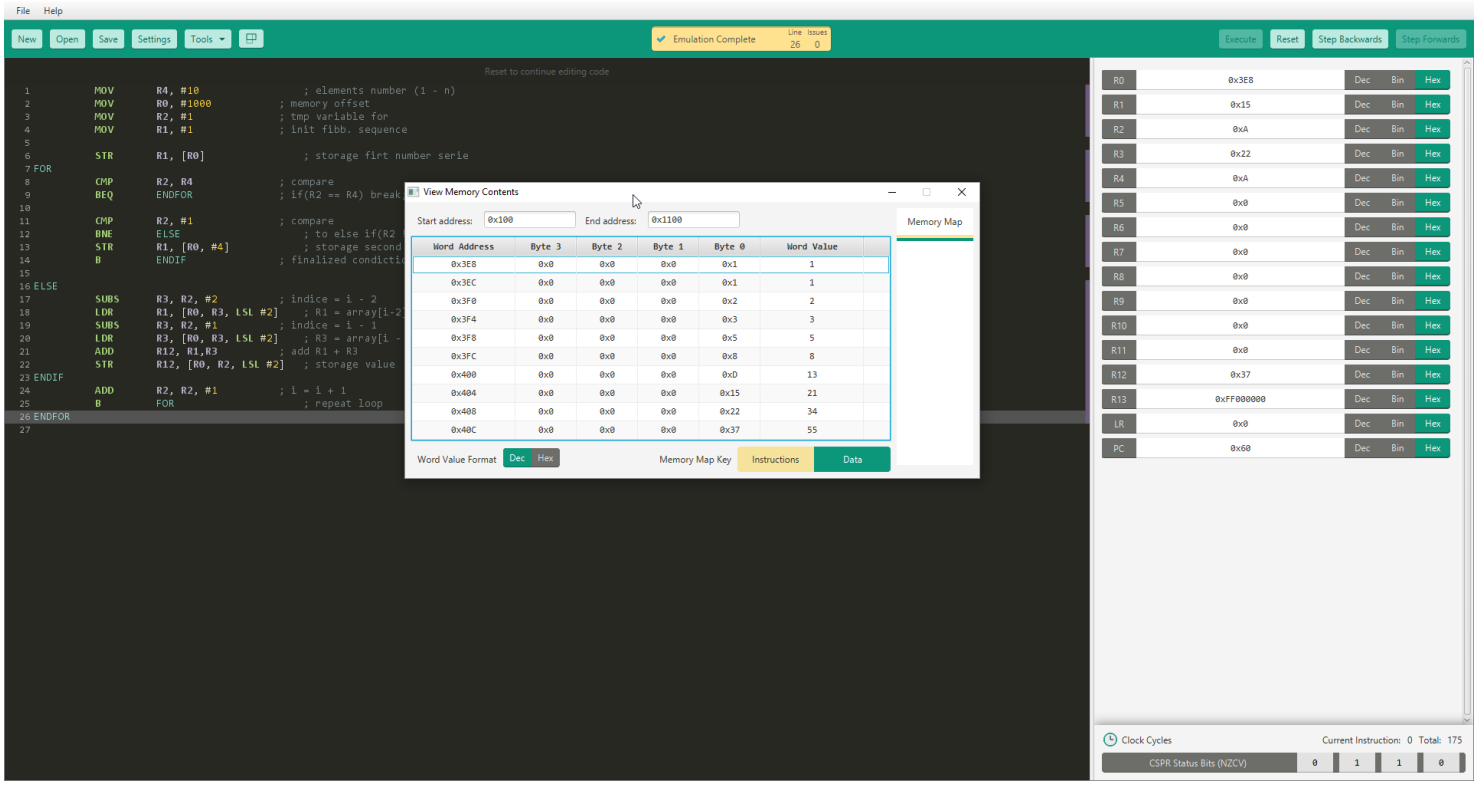
Sequência de Fibonacci

```
1.          MOV     R4, #10             ; elements number (1 - n)
2.          MOV     R0, #1000           ; memory offset
3.          MOV     R2, #1              ; tmp variable for
4.          MOV     R1, #1              ; init fibb. sequence
5.
6.          STR     R1, [R0]                ; storage firt number serie
7.  FOR
8.          CMP     R2, R4              ; compare
9.          BEQ     ENDFOR              ; if(R2 == R4) break;
10.
11.         CMP     R2, #1              ; compare
12.         BNE     ELSE                    ; to else if(R2 != 1)
13.         STR     R1, [R0, #4]            ; storage second number
14.         B       ENDIF               ; finalized condictional
15.
16. ELSE
17.         SUBS    R3, R2, #2          ; indice = i - 2
18.         LDR     R1, [R0, R3, LSL #2]    ; R1 = array[i-2]
19.         SUBS    R3, R2, #1          ; indice = i - 1
20.         LDR     R3, [R0, R3, LSL #2]    ; R3 = array[i - 1]
21.         ADD     R12, R1,R3          ; add R1 + R3
22.         STR     R12, [R0, R2, LSL #2]   ; storage value
23. ENDIF
24.         ADD     R2, R2, #1          ; i = i + 1
25.         B       FOR                     ; repeat loop
26. ENDFOR
```

https://pastebin.com/FJg3PxC6



PG

```
1.          MOV     R0, #0x100  ; 20161ceca70326
2.          ORR     R0, R0, #0x46
3.
4.          MOV     R1, #1      ; increment for
5.          MOV     R4, #10     ; elements number
6.          MOV     R2, #1000   ; offset memory
7.
8.          STR     R0, [R2]    ; storage first number serie
9.
10.
11. FOR
12.         CMP     R1,R4
13.         BEQ     ENDFOR
14.
15.         LSL     R0, R0, #1
16.         STR     R0, [R2, R1, LSL #2]
17.
18.
19.         ADD     R1, R1, #1
20.         B       FOR
21. ENDFOR
```

https://pastebin.com/PfUFJepg

Professor não coloquei no código a função LDR, no exemplo acima não ia ser muito útil a utilização do mesmo.