Spring 2022, Homework 2 (20 points in total)

Q1. (4 pts) Application Program Status Register (APSR)'s flags

After the following piece of instructions is executed, what value will be maintained in each of NZCV flags in APSR?

```
MOV     R0, #0x80000000
MOV     R1, #0x1
SUBS    R2, R1, R0
```

Flag	Value
N	1
Z	
C	
V	

Q2. (6 pts) Memory Endianness and Alignment

1) As you see the following example with #1234 at memory address 0x20000000, allocate #9876543210 to memory address 0x20001000. (2pts)

An example: 0x4D2

Big endian

Address	Data Contents (in hex)
0x20000000	04
0x20000001	D2
0x20000002	
0x20000003	

Little endian

Address	Data Contents (in hex)
0x20001000	D2
0x20001001	04
0x20001002	
0x20001003	

A question you must solve:

#9876543210

Big endian

Address	Data Contents (in hex)
0x20000000	02
0x20000001	4C
0x20000002	B0
0x20000003	16
	EA

Little endian

Address	Data Contents (in hex)
0x20001000	EA
0x20001001	16
0x20001002	B0
0x20001003	4C
	02

9876543210

16 | 617283950 | 10 A
16 | 38580246 | 14 E
16 | 2411265 | 6
16 | 150704 | 1
16 | 9419 | 0
16 | 588 | 11 (B)
16 | 36 | 12 (C)
16 | 2 | 4
11 | 0 | 2

24CB016EA

2) As you see the following example with exampleData, allocate myData to the memory and fill out the spaces to indicate how each data element is mapped. Assume that the memory is based on a 32 -bit addressing system. (2pts)

An example:

```
struct exampleData {
   char a;
   short b;
};
```

	+ 0th	+ 1st	+ 2nd	+ 3rd
0th byte	a		b	b
4th byte				
8th byte				

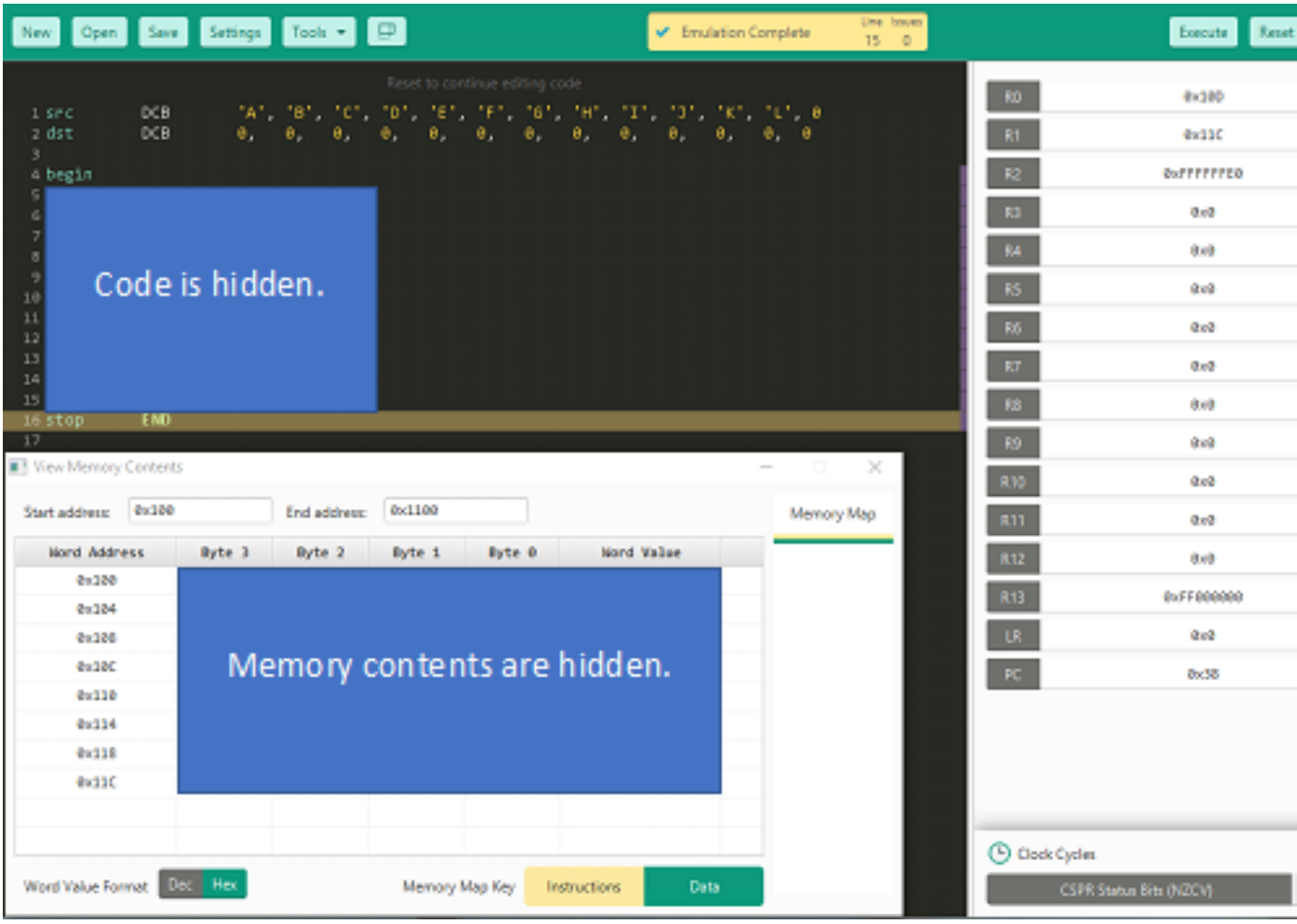
A question you should solve:

```
struct myData {
   char a; 1
   long long int b; 8
   double c; 8
   short d; 2
   char *e; 4
   float f; 4
};
```

	+ 0th	+ 1st	+ 2nd	+ 3rd
0th byte	a			
4th byte	b	b	b	b
8th byte	b	b	b	b
12th byte	c	c	c	c
16th byte	c	c	c	c
20th byte	d	d		
24th byte	e	e	e	e
28th byte	f	f	f	f
32nd byte				

Q3. (10 pts) Introduction to VisUAL

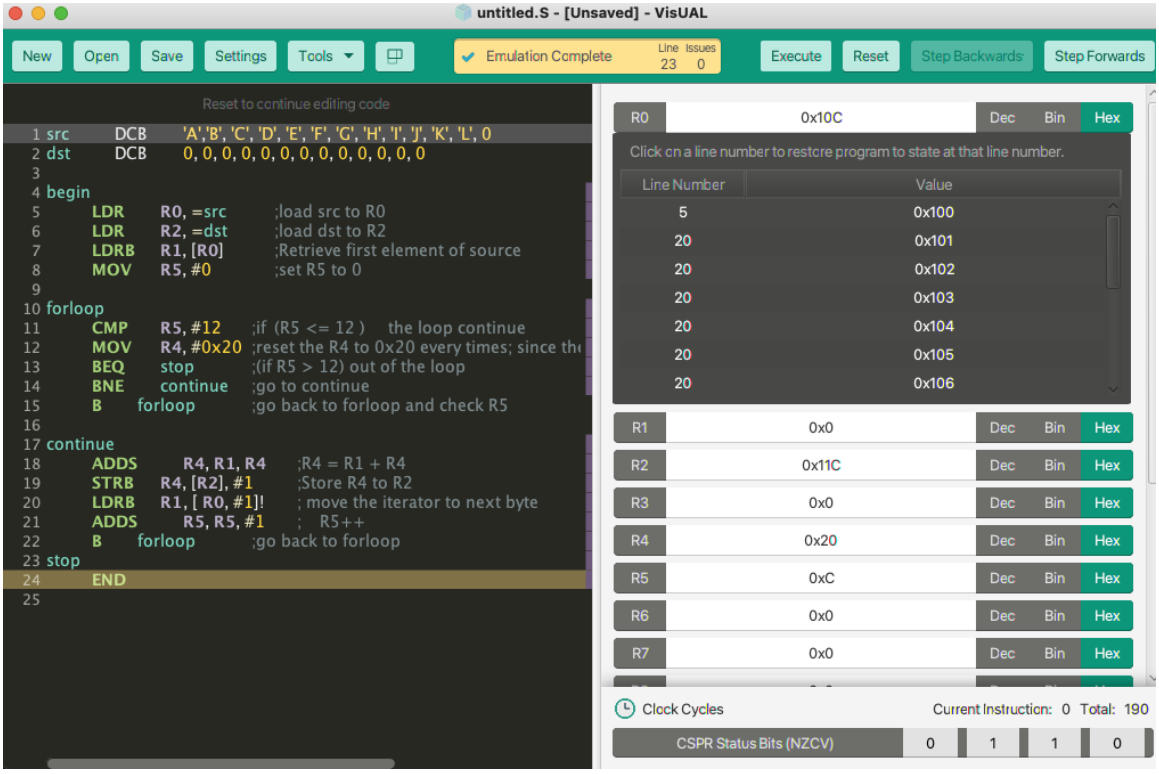
Complete the following assembly program that 1) read characters (i.e., 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L') given at address "src", 2) change them to the corresponding lower-cased characters, and 3) store them at address "dst".

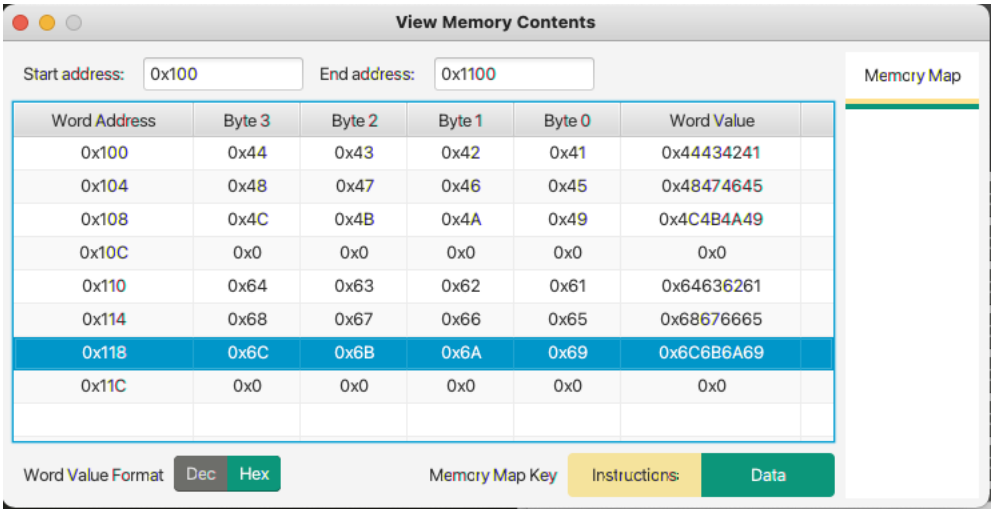


Submission: You need to submit **two files**.

File 1 (5pts): Your HW2-Q3.s.
File 2 (5pts): A **pdf** file with:

1. (1pts) A screenshot of your code.
2. (4pts) A screenshot of VisUAL's "View Memory Contents" window.


untitled.S - [Unsaved] - VisUAL
New
Open
Save
Settings
Tools
Emulation Complete
Line 23
Issues 0
Execute
Reset
Step Backwards
Step Forwards
Reset to continue editing code
1 src DCB 'A','B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 0
2 dst DCB 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
3
4 begin
5 LDR R0, =src ;load src to R0
6 LDR R2, =dst ;load dst to R2
7 LDRB R1, [R0] ;Retrieve first element of source
8 MOV R5, #0 ;set R5 to 0
9
10 forloop
11 CMP R5, #12 ;if (R5 <= 12) the loop continue
12 MOV R4, #0x20 ;reset the R4 to 0x20 every times; since th
13 BEQ stop ;(if R5 > 12) out of the loop
14 BNE continue ;go to continue
15 B forloop ;go back to forloop and check R5
16
17 continue
18 ADDS R4, R1, R4 ;R4 = R1 + R4
19 STRB R4, [R2], #1 ;Store R4 to R2
20 LDRB R1, [R0, #1]! ; move the iterator to next byte
21 ADDS R5, R5, #1 ; R5++
22 B forloop ;go back to forloop
23 stop
24 END
25
R0 0x10C Dec Bin Hex
Click on a line number to restore program to state at that line number.
Line Number Value
5 0x100
20 0x101
20 0x102
20 0x103
20 0x104
20 0x105
20 0x106
R1 0x0 Dec Bin Hex
R2 0x11C Dec Bin Hex
R3 0x0 Dec Bin Hex
R4 0x20 Dec Bin Hex
R5 0xC Dec Bin Hex
R6 0x0 Dec Bin Hex
R7 0x0 Dec Bin Hex
Clock Cycles
Current Instruction: 0 Total: 190
CSPR Status Bits (NZCV) 0 1 1 0


View Memory Contents
Start address: 0x100
End address: 0x1100
Memory Map
Word Address Byte 3 Byte 2 Byte 1 Byte 0 Word Value
0x100 0x44 0x43 0x42 0x41 0x44434241
0x104 0x48 0x47 0x46 0x45 0x48474645
0x108 0x4C 0x4B 0x4A 0x49 0x4C4B4A49
0x10C 0x0 0x0 0x0 0x0 0x0
0x110 0x64 0x63 0x62 0x61 0x64636261
0x114 0x68 0x67 0x66 0x65 0x68676665
0x118 0x6C 0x6B 0x6A 0x69 0x6C6B6A69
0x11C 0x0 0x0 0x0 0x0 0x0
Word Value Format Dec Hex
Memory Map Key Instructions Data