Assignment 2: Computer Organization

In the assembly C code below, $f0, f1, N$ and fN are C variables.
Analyse it and answer questions $1 - 3$.

```
1               void main()
2               {
3               int f0 = 0, f1 = 1, N = 9, fN;
4               N− = 1;
5               __asm{
6                       MOV ECX, N
7                       MOV EBX, f0
8                       MOV EAX, f1
9       LBL:            ADD EAX, EBX
10                      SUB EBX, EAX
11                      NEG EBX
12                      DEC ECX
13                      JNZ LBL
14                      MOV fN, EAX
15              }
```

(1) **Qn.1** NUMERICAL

What is the value in EAX when the program reaches statement 14?

(2) **Qn.2** NUMERICAL

If the instructions were executing on a 32-bit processor, what is the largest value one can assign to N in statement 3 such that the program reaches statement 14 without an overflow of the accumulator in statement 9?

(3) **Qn.3** NUMERICAL

Let the answer to question Qn.2 be k. If we replace $N = 9$(in statement 3) by $N = k$, what is the value in EAX when the program reaches statement 14?

(4) **Qn.4** MULTIPLE CHOICE   One answer only

(1 point)
Say the opcode size of REPNE SCASB is 1 byte and at present EIP=0x0F21 and ECX=10. The instruction REP MOVSB is executed. After the execution, we observe that CX=03. Then, EIP=_____?

a. 0x0F22
b. 0x0F29
c. 0x0F28
d. 0x0F2B

(5) **Qn.5** MULTIPLE CHOICE   One answer only

(2 points)

NOTE:- ECS=0x0141, EDS=0x1600, ESS=0x2512, EES=0x3509, EAX=0x0116, EBP=0x4567.

Assume that there are two arrays as follows:-

Address-Array$_1$	Data-Array$_1$	Address-Array$_2$	Data-Array$_2$
0x16100	11	0x35290	00
0x16101	14	0x35291	00
0x16102	17	0x35292	00
0x16103	19	0x35293	00
0x16104	26	0x35294	00

After the following snipped code executes, what will be the values in Array-1, Array-2?

```
CLD
MOV ESI, 0x102
MOV EDI, 0x200
MOV ECX, 3
REP MOVSB
```

a.

Address-Array$_1$	Data-Array$_1$	Address-Array$_2$	Data-Array$_2$
0x16100	11	0x35290	11
0x16101	14	0x35291	14
0x16102	17	0x35292	17
0x16103	19	0x35293	00
0x16104	26	0x35294	00

b.

Address-Array$_1$	Data-Array$_1$	Address-Array$_2$	Data-Array$_2$
0x16100	00	0x35290	17
0x16101	00	0x35291	19
0x16102	17	0x35292	26
0x16103	19	0x35293	00
0x16104	26	0x35294	00

c.

Address-Array$_1$	Data-Array$_1$	Address-Array$_2$	Data-Array$_2$
0x16100	11	0x35290	17
0x16101	14	0x35291	19
0x16102	00	0x3529	26
0x16103	00	0x35293	00
0x16104	00	0x35294	00

d.

Address-Array$_1$	Data-Array$_1$	Address-Array$_2$	Data-Array$_2$
0x16100	11	0x35290	11
0x16101	14	0x35291	14
0x16102	17	0x35292	17
0x16103	19	0x35293	19
0x16104	26	0x35294	26

2

Answer the following questions 6-8 based on the code snippet shown below: (Assume ECS=0x0251, EDS=0x4106, ESS=0x6247, EES=0xA270)

1	STD
2	MOV AL, 14
3	MOV ECX, 9
4	MOV EDI, 0x0216
5	REPNE SCASB

Address	Data
0xA2913	18
0xA2914	14
0xA2915	18
0xA2916	21
0xA2917	14

(6) **Qn.6** MULTIPLE CHOICE   One answer only

(2 points)
The value stored in the ECX register after the above code snippet executes:

   a. 5
   b. 6
   c. 7
   d. 8

(7) **Qn.7** MULTIPLE CHOICE   One answer only

(2 points)
The value stored in the register EDI after the above code snippet executes:

   a. 0x0217
   b. 0x0216
   c. 0x0213
   d. 0x0214

(8) **Qn.8** MULTIPLE CHOICE   One answer only

(1 point)
What is the value of the Direction Flag after the above code snippet executes

   a. Cannot say
   b. 0
   c. 1

Consider the following for questions 9-12:

You are given an 8 bit microprocessor with three registers general purpose registers **A, B and C**, an instruction pointer **IP** and the following instruction set:

Mnemonic	Functionality	Comment
MOV $R1, R2$	$R1 = R2$	Copy contents of Register $R1$ to $R2$
MOV $R1, \#N$	$R1 = N$	Copy a constant N (in hex) to $R1$
ADD $R1, R2$	$R1 = R1 + R2$	Addition
CALL $\#FN\_ADDR$	$IP = FN\_ADDR$	Subroutine call – also PUSHES the return address on stack
RET	$IP = TOS$	Top of stack loaded into **IP**
END		Suspends execution of the program with the **IP** unaltered

The following Assembly Program is loaded in the code segment and executed with the instruction pointer initialized to the address **0xF0**.

ADDRESS	INSTRUCTION
0xE0	MOV C, B
0xE1	MOV B, A
0xE2	MOV A, C
0xE3	RET
0xF0	MOV A, #2
0xF2	MOV B, #26
0xF4	CALL #E0
0xF6	ADD C, A
0xF7	END

Answer the following questions about the value of registers at the end of the program. Enter your answers in Decimal (base-10)

(9) **Qn.9** NUMERICAL

What is the value of **IP**?

(10) **Qn.10** NUMERICAL

What is the value of register **A**?

(11) **Qn.11** NUMERICAL

What is the value of register **B**?

(12) **Qn.12** NUMERICAL

What is the value of register **C**?

Total of marks: 20