

Blank page for working

CSE1303 Part B — Introduction to Computer Systems

Question B1 [0.5 + 1 + 0.5 = 2 marks]

This question is about integer representation.

a. Consider the following piece of C code:

```
int n;  
scanf("%d", &n); /* %d is decimal */  
n = n + 37;  
printf("%x\n", n); /* %x is hexadecimal */
```

How does the computer represent n in memory? Circle the correct answer:

- (a) binary (unsigned)
- (b) binary (two's complement)
- (c) decimal
- (d) hexadecimal
- (e) none of the above

b. Convert the decimal number 133 into 8-bit *unsigned* binary. *Show all working.*

c. Write your above answer to (b) in **both hexadecimal and octal.**

2

Blank page for working

Question B2 [1 + 1 = 2 marks]

This question is about 8-bit *signed* integer arithmetic.

a. Perform the following binary addition. *Show all working.*

$$\begin{array}{r} 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0 \\ +\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1 \\ \hline \end{array}$$

b. Perform the following binary subtraction. *Show all working.*

$$\begin{array}{r} 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0 \\ -\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1 \\ \hline \end{array}$$

2

Blank page for working

Question B3 [1 + 1 + 1 + 1 = 4 marks]

This question is about floating-point representation. The following table shows the field sizes for a C float and may be useful for this question:

Sign	Exponent (127 excess)	Mantissa (excluding implicit 1)
1 bit	8 bits	23 bits

A certain four-byte section of memory contains the following values, shown here in hexadecimal:

00	lowest address
00	
A0	
C2	highest address

a. Assuming a little-endian computer, what 32-bit value does this represent? Write your answer in **both hexadecimal and binary**.

b. If this value is a C float, write in the space below the bits that comprise its sign, exponent and mantissa. Write your answer in **binary**.

--	--	--

c. Fill in the gaps below to show this number's value.

		× 2		₁₀
sign (+/-)	mantissa (write in <i>binary</i> ; show binary point)		exponent (write in <i>decimal</i>)	

d. What is the above value expressed in decimal?

4

Blank page for working

Question B4 [0.5 + 0.5 + 0.5 + 0.5 = 2 marks]

This question is about pointers and arrays. Assume for this question that addresses are 32 bits in size.

The C sizeof operator returns the number of bytes that a variable takes up in memory.

Given the following C code:

```
char s[20] = "This is a string";  
char *p = s;
```

What would be printed by the following statements?

a. `printf("%d\n", sizeof s);`

b. `printf("%d\n", sizeof p);`

c. `printf("%d\n", strlen(s));`

d. `printf("%d\n", strlen(p));`

2

Blank page for working

END OF TEST