Unit 1: Pointers, Memory, and Number Representation

For this problem we are using a 64-bit x86-64 machine (**little endian**). The current state of memory (values in hex) is shown below:

Word Addr	+0	+1	+2	+3	+4	+5	+6	+7
0x30	93	DC	B8	7A	3B	1A	B2	0C
0x38	D3	A6	A4	71	E2	23	9C	59
0x40	60	15	68	76	D3	E6	25	BE
0x48	A4	A5	DB	BE	56	AF	D1	2E
0x50	17	1F	95	C4	24	63	D2	62
0x58	B1	7A	44	58	C7	C4	03	81

```
int* ip = 0x4C;
short* sp = 0x36;
```

(A) Using the values shown above, fill in the C type and hex value for each of the following C expressions.

C Expression	C Type	Hex Value
`ip + 3`		
`sp[-1] + 1`		
`*((char*)ip - 1)`		

(B) Let the variable **`signed char x`** be located at the address **`0x44`** in the memory diagram above.

(i) What is the value of x in decimal?

(ii) For each of the following expressions, indicate whether it will result in a positive, negative, or zero result.

`x << 2`	Positive	Negative	Zero
`!(x ^ 0xD4)`	Positive	Negative	Zero
`x >> 1`	Positive	Negative	Zero

(iii) Find the *smallest 8-bit unsigned numeral* c (answer in hex) such that **`c + x`** causes unsigned over *but not* signed overflow in 8 bits.