

1. Number Conversion

Fill in the table below

Binary	Octal	Decimal	Hexadecimal
	154		
110 0101 1101			
			0xBED
		2014	
	2014		

2. Operations

(1) Given A and B with hexadecimal expression 0xED and 0x3F respectively. Calculate the values of the following expressions

- a) A & B
- b) A && B
- c) A | B
- d) A || B
- e) A ^ B
- f) ~A | ~B

(2) Fill in the table below with the results of shift operation given below (Assume X is 8-bit wide) . Please answer in hexadecimal.

X	X<<4	X>>3(Logical)	X>>3(Arithmetic)
0x5D			
0xC9			
0xA2			
0xBF			

(3) Design C expressions using bitwise (& | ~ ^) and logical (&& || !) operators, which return 1 under the specific conditions described below, otherwise return 0.

- A. all bits of x are 1.
- B. all bits of x are 0.
- C. bits of x's least significant byte are 1.
- D. bits of x's least significant byte are 0.

(4) Design a C expression, which generates a word (32-bit) consisting of the lower 16 bits of x and the remaining bits of y.

For example, x = 0x89ABCDEF and y = 0x76543210, it will generate 0x7654CDEF

3. Byte Ordering

(1) Consider the following function:

```
typedef unsigned char *byte_pointer;
```

```
void show_bytes(byte_pointer start, int len)
```

```

{   int i;
    for (i = 0; i < len; i++)
        printf("%.2x", start[i]);
    printf("\n");
}

```

```

int val = 0x28AE3514
byte_pointer valp = (byte_pointer) &val;

```

What is the output of the following call to show_bytes on big-endian and little-endian machines respectively?

	little-endian	big-endian
show_bytes(valp, 1);		
show_bytes(valp, 2);		
show_bytes(valp, 3);		
show_bytes(valp, 4);		