

Name :

UID :

1.

Assume:

```
int x = rand();
```

```
int y = rand();
```

```
unsigned ux = (unsigned) x;
```

Are the following statements always true?

a.

```
ux >> 3 == ux/8
```

b.

```
given x > 0,
```

```
((x << 5) >> 6) > 0
```

c.

```
~x + x >= ux
```

d.

```
given x & 15 == 11,
```

```
x & 0000...1111 == 0000...1011
```

```
( ~(x >> 3) & (x >> 2)) << 31) >= 0
```

e.

```
given ((x < 0) && (x + x < 0))
```

```
x + ux < 0
```

f.

```
given ((x < 0) && (y < 0) && (x + y > 0))
```

```
((x | y) >> 30) == -1
```

2.

Given: x has a 4 byte value of 255

0x000000FF

What is the value of the byte with the lowest address in a

a. big endian system?

b. little endian system?

3.

Endianness

a. Suppose we declared the following 4 byte int:

```
int x = 254;
```

and we stored this in memory location 0x100 on a little-endian system. What values would be stored in the following memory locations?

0x100	0x101	0x102	0x103

b. Suppose we declared an array of ints:

```
int arr[] = {1, 2};
```

and we stored this in memory location 0x100 on a little endian system. What values would be stored in the following memory locations?

0x100	0x101	0x102	0x103	0x104	0x105	0x106	0x107

c. Suppose we declared a string:

```
char * s = "hello";
```

and we stored this in memory location 0x100 on a little endian system. What values would be stored in the following memory locations?

note: it's a good idea to get familiar with hex encodings of alphabetical characters, but for convenience, the hexadecimal encodings of the characters are: h (0x68), e (0x65), l (0x6c), and o (0x6f)

0x100	0x101	0x102	0x103	0x104	0x105