COMP 122/L
Summer 2023

Bitwise Operations (Answers)

All answers should be in 8-bit binary.

1.) What is `1011 1100 << 3`?

`1110 0000`

2.) What is `0110 1101 >> 2`, for logical shift right?

`0001 1011`

3.) What is `0110 1101 >> 2`, for **arithmetic** shift right?

`0001 1011`

4.) What is `1110 0010 >> 4`, for logical shift right?

`0000 1110`

5.) What is `1110 0010 >> 4`, for **arithmetic** shift right?

`1111 1110`

6.) What is:

```
  11001110
& 10110101
  10000100
```

7.) What is:

```
  11000001
| 10110101
  11110101
```

8.) What is:

```
  11001110
^ 10110100
  01111010
```

9.) Assume you have an unknown 8-bit number. Specify the bitmask and operation needed to **isolate** the bit in position 6. The result of the mask and the operation should be all zeros if the bit in position 6 is a 0, and the result should be non-zero otherwise.

```
have: XXXX XXXX
want: 0X00 0000

mask: 0100 0000, with &.  Keep in mind that X & 0 = 0, whereas X
& 1 = X.
```

10.) Assume you have an unknown 8-bit number. Specify the bitmask and operation needed to **set** the bit in position 6. The result of the mask and the operation should be the same as the original number, except the bit in position 6 will always be set (1).

```
have: XXXX XXXX
want: X1XX XXXX

mask: 0100 0000, with |.  Keep in mind that X | 0 = X, whereas X
| 1 = 1.
```

11.) Assume you have an unknown 8-bit number. Specify the bitmask and operation needed to **unset** the bit in position 6. The result of the mask and the operation should be the same as the original number, except the bit in position 6 will always be unset (0).

```
have: XXXX XXXX
want: X0XX XXXX

mask: 1011 1111, with &.  Keep in mind that X & 1 = X, whereas X
& 0 = 0.
```