

1 Bit Manipulation

1. Write 22 in binary.
2. Assuming x_1, x_2, \dots, x_n are integers. What is $(x_1 \wedge x_2 \wedge \dots \wedge x_n) \wedge (x_1 \vee x_2 \vee \dots \vee x_n)$?
3. Write an expression to check whether a 32-bit integer is less than 0 using only `==` and the bit operators.
4. What does the following code do?

```
public static int mysteryBit(int n) {  
    return n & (n - 1);  
}
```

5. Write a program to count the number of 1 bits in an integer. You can use the function in part 4 as a hint.

```
public static int countBits(int n) {  
  
  
  
}
```

2 Algorithmic Analysis

1. For each of the following function, find the Big-Theta expression for:
 - a) The number of `i += 1` or `i *= 2` operations
 - b) The number of `j += 1` operations
 - c) The number of print operations
 - d) The runtime of the function

```
public static void printIndices(int n) {  
    for (int i = 0; i < n; i += 1) {  
        for (int j = 0; j < i; j += 1) {  
            System.out.println(i + j);  
        }  
    }  
}
```

```
public static void printIndices2(int n) {  
    for (int i = 1; i < n; i *= 2) {  
        for (int j = 0; j < i; j += 1) {  
            System.out.println(j);  
        }  
    }  
}
```

2. What is the big-Theta running time of the following functions?

```
public int weirdFib(int n) {
    if (n <= 1) {
        return n;
    }
    return weirdFib(n - 1) + weirdFib(n - 1);
}
```

```
public static void mystery(int n) {
    if (n == 1) {
        return;
    }
    for (int i = 0; i < n; i += 1) {
        mystery(n-1);
    }
}
```

3 Regex

Write a Java regular expression to match each of the following sets of binary strings. You may only use the following characters: `()|01*`

- 1) All binary strings
- 2) Binary strings that begins and ends with 1
- 3) Binary strings that contains at least three 1s
- 4) Binary string that contains at least three consecutive 1s
- 5) Binary string that doesn't contain the substring 110.