2.7

Program 1	O(n)
n	Runtime(ns)
10	200
100	200
1000	1600
10000	18500
Program 2	**O(n^2)**
n	Runtime(ns)
10	300
100	16700
1000	1564000
10000	160669800
Program 3	**O(n^3)**
n	Runtime(ns)
10	1800
100	1791700
1000	1877968100
Program 4	**O(n^2)**
n	Runtime(ns)
10	300
100	8000
1000	779300
10000	79782700
Program 5	**O(n^5)**
n	Runtime(ns)
10	12900
100	1521842000
Program 6	**O(n^4)**
n	Runtime(ns)
10	2700
100	19735200

```
if (number == 1)
{
    for (int i = 0; i < n; i++)
        sum++;
}
if (number == 2)
{
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
            sum++;
}
if (number == 3)
{
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n * n; j++)
            sum++;
}
if (number == 4)
{
    for (int i = 0; i < n; i++)
        for (int j = 0; j < i; j++)
            sum++;
}
if (number == 5)
{
    for (int i = 0; i < n; i++)
        for (int j = 0; j < i * i; j++)
            for (int k = 0; k < j; k++)
                sum++;
}
if (number == 6)
{
    for (int i = 1; i < n; i++)
        for (int j = 1; j < i * i; j++)
            if (j % i == 0)
                for (int k = 0; k < j; k++)
                    sum++;
}
```

Program 1: The runtime is O(n) and the actual runtime appears to increase in a linear rate
Program 2: The runtime is O(n^2) and the actual runtime appears to increase in a quadratic rate
Program 3: The runtime is O(n^3) and the actual runtime appears to increase in a cubic rate
Program 4: The runtime is O(n^2) and the actual runtime appears to increase in a quadratic rate
Program 5: The runtime is O(n^5) and the actual runtime appears to increase at a rate of the power of 5.
Program 6:The runtime is O(n^4) and the actual runtime appears to increase to the fourth power.

2.11

a. 2.5 ms
b. 3.37 ms
c. 12.5 ms
d. 62.5 ms

2.12

a. 12,000,000
b. 912,192
c. 34,641
d. 4,932