Complexity Analysis

Yu-Shuen Wang, CS, NCTU

Program running time

When is the running time (waiting time for user) noticeable/important?

Program running time – Why?

When is the running time (waiting time for user) noticeable/important?

- web search
- database search
- real-time systems with time constraints

Factors that determine running time of a program

- problem size: n
- basic algorithm / actual processing
- memory access speed
- CPU/processor speed
- # of processors?
- compiler/linker optimization?

Running time of a program or transaction processing time

- amount of input: n → min. linear increase
- basic algorithm / actual processing → depends on algorithm!
- memory access speed → by a factor
- CPU/processor speed → by a factor
- # of processors? → yes, if multi-threading or multiple processes are used.
- compiler/linker optimization? → ~20%

Time Complexity

- measure of algorithm efficiency
- has a big impact on running time.
- Big-O notation is used.
- To deal with n items, time complexity can be $O(1)$, $O(\log n)$, $O(n)$, $O(n \log n)$, $O(n^2)$, $O(n^3)$, $O(2^n)$, even $O(n^n)$.

Coding example #1

```
for ( i=0 ; i<n ; i++ )
    m += i;
```

Coding example #2

```
for ( i=0 ; i<n ; i++ )
        for( j=0 ; j<n ; j++ )
             sum[i] += entry[i][j];
```

Coding example #3

```
for ( i=0 ; i<n ; i++ )
        for( j=0 ; j<i ; j++ )
            m += j;
```

Coding example #4

```
i = 1;
while (i < n) {
    tot += i;
    i = i * 2;
}
```

Coding example #5

```
for ( i=0 ; i<n ; i++ )
   for( j=0 ; j<n ; j++ )
        for( k=0 ; k<n ; k++ )
            sum[i][j] += entry[i][j][k];
```

Coding example #6

```
for ( i=0 ; i<n ; i++ )
        for( j=0 ; j<n ; j++ )
             sum[i] += entry[i][j][0];

for ( i=0 ; i<n ; i++ )
        for( k=0 ; k<n ; k++ )
             sum[i] += entry[i][0][k];
```

Coding example #7

```
for ( i=0 ; i<n ; i++ )
        for( j=0 ; j< sqrt(n) ; j++ )
            m += j;
```

Coding example #8

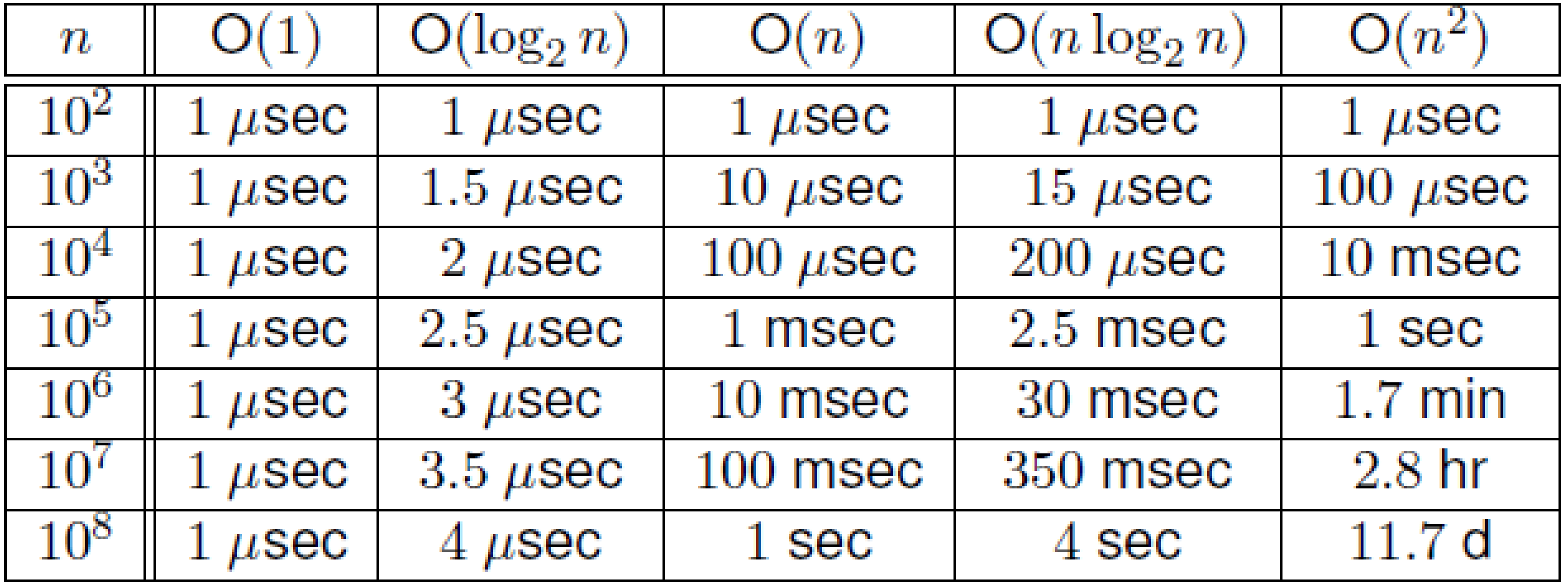
```
for ( i=0 ; i<n ; i++ )
       for( j=0 ; j< sqrt(995) ; j++ )
           m += j;
```

Coding example #9

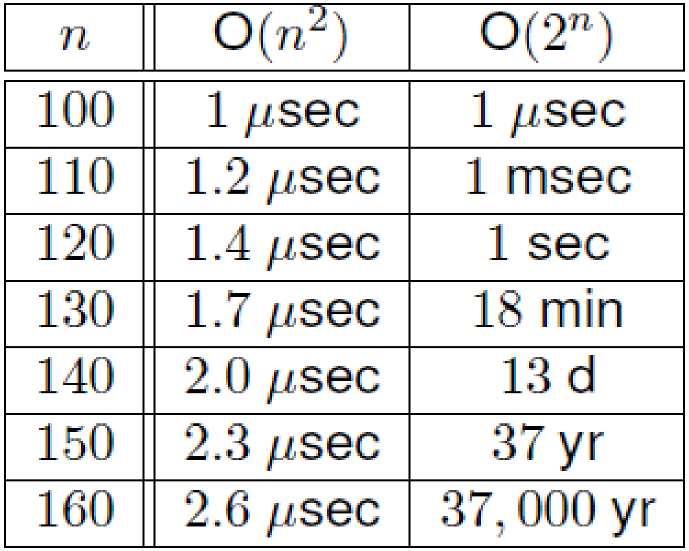
```
int total(int n)
   for( i=0 ; i < n; i++)
    subtotal += i;

main()
   for ( i=0 ; i<n ; i++ )
    tot += total(i);
```

Compare running time growth rates



n	$O(1)$	$O(\log_2 n)$	$O(n)$	$O(n \log_2 n)$	$O(n^2)$
10^2	1 μsec	1 μsec	1 μsec	1 μsec	1 μsec
10^3	1 μsec	1.5 μsec	10 μsec	15 μsec	100 μsec
10^4	1 μsec	2 μsec	100 μsec	200 μsec	10 msec
10^5	1 μsec	2.5 μsec	1 msec	2.5 msec	1 sec
10^6	1 μsec	3 μsec	10 msec	30 msec	1.7 min
10^7	1 μsec	3.5 μsec	100 msec	350 msec	2.8 hr
10^8	1 μsec	4 μsec	1 sec	4 sec	11.7 d



n	$O(n^2)$	$O(2^n)$
100	1 μsec	1 μsec
110	1.2 μsec	1 msec
120	1.4 μsec	1 sec
130	1.7 μsec	18 min
140	2.0 μsec	13 d
150	2.3 μsec	37 yr
160	2.6 μsec	37,000 yr