Date |
Page |

(2)

1 start

```
for(i=0; i<n; i++)            - n+1
{
    for(j=0; j<n; j++)        - n(n+1)
    {
        stmt;                 - n×n
    }
}
```

end

$\underline{\underline{O(n^2)}}$

2 start

```
for(i=0; i<n; i++)
{
    for(j=0; j<i; j++)
    {
        stmt;
    }
}
```

end

$f(n) = \frac{n^2+n}{2}$

$\underline{\underline{O(n^2)}}$

- -

Date |

Page |

(3)

1: start

```
    p=0;
    for( i=1; p<=n; i++)
    {
        p=p+i;
    }
end
```

Assume $p > n$

$\because p = \frac{k(k+1)}{2}$

$\frac{k(k+1)}{2} > n$

$k^2 > n$

$k > \sqrt{n}$

$\underline{O(\sqrt{n})}$

---

(4)

1: start

```
    for ( i=1; i<n; i=i*2)
    {
        stmt;
    }
end
```

i will run till 2^k times

Assume $i >= n$

$\therefore i = 2^k$

$2^k >= n$

$2^k = n$

$k = \log_2 n$

$O(\log_2 n)$

(5)

1. start

```
for(i=n; i>=1; i=i/2)
{
    stmt;
}
```

end

assume $i<1$

$\therefore \frac{n}{2^k} < 1$

$\frac{n}{2^k} = 1$

$n = 2^k$

$k = \log_2 n$

---

(6)

1. start

```
for (i=0; i*i<n; i++)
{
    stmt;
}
```

end

$i*i < n$

$i^2 >- n$

$i^2 = n$

$i = \sqrt{n}$

$O(\sqrt{n})$

```
start
    for(i=0; i<n; i++)
    {
        stmt;
    }                         - n
    for(j=0; j<n; j++)
    {
        stmt;                 - n
    }                         ----
end                           2n
            O(n)
```

```
start
    p=0
    for(i=1; i<n; i=i*2)
    {
        p++;    - p = log n
    }
    for(j=1; j<p; j=j*2)
    {
        stmt;   - log p
    }
                O(log log n)
```

Date |
Page |

(9)

! start

```
for(i=0; i<n; i++)            - n
{
    for(j=1; j<n; j=j*2)      - n x logn
    {
        stmt;                 - n x logn
    }
}
```

$2n\log n + n$

$O(n\log n)$

- for(i=0 ; i<n ; i++) $- O(n)$
- for(i=0; i<n; i=i+2) $-\frac{n}{2} - O(n)$
- for(i=n; i>1 ; i--) $- O(n)$
- for(i=1; i<n; i=i*2) $- O(\log_2 n)$
- for(i=1; i<n; i=i*3) $- O(\log_3 n)$

→ Analysis of if and while

- start

```
    i = 0              - 1
    while (i < n)      - n+1
    {
        stmt;          - n
        i++;           - n
    }
end
```

$f(n)$ $3n+2$

$O(n)$

- start

```
    a = 1
    while (a < b)
    {
        stmt;
        a = a * 2
    }
end
```

terminate

$a \geq b$

$\because a = 2^k$

$2^k \geq b$

$2^k = b$

$k = \log_2 b$

$O(\log n)$

- start

```
    i = n
    while (i > 1)
    {
        stmt;
        i = i/2)
    }
```

$O(\log_2 n)$