

Complexity Analysis:

Problem1:

```
for ( i=0; i<n; i++)  ----- n+1
{
  for ( j=0; j<n; j++)    n*(n+1)
  {
    statement;          n*n=n2
  }
}
```

$$T(n) = n+1 + n*(n+1) + n^2 \Rightarrow 2n+1+2n^2 \quad 2(n^2+n)+1 \Rightarrow O(n^2) \text{ Showed}$$

Problem2:

```
for ( i=1; i<n; i+2)  {
    statement;  ----- n/2
}
```

$$T(n) = O(n/2) \Rightarrow O(1/2*n) = O(n) \text{ Showed}$$

Problem3:

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

$$T(n) = 1+2+3+\dots+n = n(n+1)/2$$

$$= \frac{1}{2}(n^2+n) \Rightarrow O(n^2) \text{ showed}$$

i	j	no. of time
0	0	0
1	0	1
2	0	2
	1	
	2	×
3	0	3
	1	
	2	
	3	×
.....		
.....		
n		n

Problem4: $p=0;$ for ($i=1; p \leq n; i++$) { $p=p+i;$ } Therefore, $p = \frac{1}{2}(k^2+k);$ Assume $p > n$ $\frac{1}{2}(k^2+k) > n \Rightarrow k^2 > n \Rightarrow k > \sqrt{n}$ $\Rightarrow O(\sqrt{n})$ showed	<table> <tr> <th>i</th><th>p</th></tr> <tr><td>1</td><td>$0+1=1$</td></tr> <tr><td>2</td><td>$1+2=3$</td></tr> <tr><td>3</td><td>$1+2+3=6$</td></tr> <tr><td>4</td><td>$1+2+3+4=10$</td></tr> <tr><td></td><td>.....</td></tr> <tr><td></td><td>.....</td></tr> <tr><td>k</td><td>$1+2+3+4+-----+k=k(k+1)/2$</td></tr> </table>	i	p	1	$0+1=1$	2	$1+2=3$	3	$1+2+3=6$	4	$1+2+3+4=10$		k	$1+2+3+4+-----+k=k(k+1)/2$
i	p																
1	$0+1=1$																
2	$1+2=3$																
3	$1+2+3=6$																
4	$1+2+3+4=10$																
																
																
k	$1+2+3+4+-----+k=k(k+1)/2$																

Problem5: for (i=1; i<n; i=i*2) { statement; } Assume i>=n i=2 ^k 2 ^k >=n k*log ₂ ² =log ₂ ⁿ => k= log ₂ ⁿ T(n)=O(log ₂ ⁿ)	<div><div>i</div><div>1</div><div>=2⁰</div><div>1x2=2¹</div><div>2¹x2=2²</div><div>2²x2=2³</div><div>.....</div><div>.....</div><div>2^{k-1}x2=2^k</div></div>
--	--

Problem6: for (i=n; i>=1; i=i/2) { statement; } Assume i<1 n/2 ^k < 1 n/2 ^k = 1 n=2 ^k k*log ₂ ² =log ₂ ⁿ => k= log ₂ ⁿ T(n)=O(log ₂ ⁿ)	<table><tr><th>i</th></tr><tr><td>n</td></tr><tr><td>n/2¹</td></tr><tr><td>n/2²</td></tr><tr><td>n/2³</td></tr><tr><td>.....</td></tr><tr><td>.....</td></tr><tr><td>n/2^k</td></tr></table>	i	n	n/2 ¹	n/2 ²	n/2 ³	n/2 ^k
i									
n									
n/2 ¹									
n/2 ²									
n/2 ³									
.....									
.....									
n/2 ^k									

Problem7: for ($i=0; i*i < n; i++$) { statement; }	$i*i < n$ $i*i \geq n$ $i^2 = n$ $i = \sqrt{n}$
--	--

$$\begin{aligned}
 \text{for}(i=0; i < n; i++) &\text{--- } O(n) \\
 \text{for}(i=0; i < n; i=i+2) &\text{--- } \frac{n}{2} \quad O(n) & \frac{n}{2} &= O(n) \\
 \text{for}(i=n; i > 1; i--) &\text{--- } O(n) & \frac{n}{200} &= O(n) \\
 \text{for}(i=1; i < n; i=i*2) &\text{--- } O(\log_2 n) \\
 \text{for}(i=1; i < n; i=i*3) &\text{--- } O(\log_3 n) \\
 \text{for}(i=n; i > 1; i=i/2) &\text{--- } O(\log_2 n)
 \end{aligned}$$

Analysis of if & while

$$\begin{aligned}
 &i=0; \text{--- } 1 \\
 &\text{while}(i < n) \text{--- } n+1 \\
 &\{ \\
 &\quad \text{stmt}; \text{--- } n \\
 &\quad i++; \text{--- } n \\
 &\} \\
 &\text{--- } \\
 &\underline{f(n) = 3n + 2} \\
 &\quad O(n)
 \end{aligned}$$

$$\begin{aligned}
 &\text{for}(i=0; i < n; i++) \text{--- } n+1 \\
 &\{ \\
 &\quad \text{stmt}; \text{--- } n \\
 &\} \\
 &\text{--- } \\
 &\underline{f(n) = 3n + 2} \\
 &\underline{f(n) = 2n + 1} \\
 &\quad O(n)
 \end{aligned}$$

Analysis of if & while

```

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

$$\begin{array}{c}
 a \\
 \hline
 1 \\
 1 \times 2 = 2 \\
 2 \times 2 = 2^2 \\
 2^2 \times 2 = 2^3 \\
 \vdots \\
 2^k
 \end{array}$$

Terminate

$$a \geq b$$

$$\therefore a = 2^k$$

$$2^k \geq b$$

$$2^k = b$$

$$k = \log_2 b$$

$$O(\log n)$$

Analysis of if & while

```

i = 1;
k = 1;
while (k < n)
{
    stmt;
    k = k + i;
    i++;
}
O(\sqrt{n})
    
```

i	k
1	1
2	1+1=2
3	2+2
4	2+2+3
5	2+2+3+4
\vdots	\vdots
\vdots	\vdots
m	2+2+3+4+...+m
	$\frac{m(m+1)}{2}$

$$k \geq n$$

$$\frac{m(m+1)}{2} \geq n$$

$$m^2 \geq n$$

$$m = \sqrt{n}$$

Analysis of if & while

<code>while(m != n)</code>	<u>m=16</u>	<u>n=2</u>	
<code>{</code>	14	2	
<code>if(m > n)</code>	12	2	
<code>m = m - n;</code>	10	2	
<code>else</code>	8	2	$\frac{16}{2}$
<code>n = n - m;</code>	6	2	
	4	2	$\frac{n}{2}$
<code>}</code>	2	2	2
	min $O(1)$		<u>$O(n)$</u>

Analysis of if & while

Algorithm Test(n)

```

{
    if(n < 5)
        printf("%d", n); — 1
    else
        for(i=0; i < n; i++)
            printf("%d", i); — n
}

```

Best — $O(1)$
 worst — $O(n)$