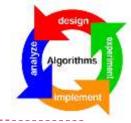
Analyzing Algorithms

Time Analysis/Complexity-Few Cases



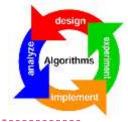
```
Case:1
for(i=0;i<n;i++)</pre>
for(i=0;i<n;i=i+2)
for(i=n;i>0;i--)
```

```
Case:2
for(i=0;i<n;i++)</pre>
        for(j=0;j<n;j++)</pre>
                 stmt;
```



Case:3

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



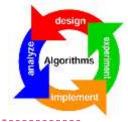
```
Case:4
for(i=1;i<n;i=i*2)
{
    stmt;
}</pre>
```

```
Case:5
for(i=n;i>=1;i=i/2)
{
    stmt;
}
```



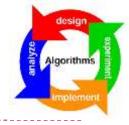
Case:6

```
p=0;
for(i=1;p<=n;i++)
{
    p=p+i;
}</pre>
```



```
Case:7
for(i=0;i*i<n;i++)
{
    stmt;
}</pre>
```

```
Case:8
for(i=0;i<n;i++)
{
    stmt;
}
for(j=0;j<n;j++)
{
    stmt;
}</pre>
```



```
Case:9
p=0;
for(i=1;i<n;i=i*2)</pre>
        p++;
for(j=1;j<p;j=j*2)</pre>
        stmt;
```

```
▶ Case:10
for(i=0;i<n;i++)</pre>
        for(j=1;j<n;j=j*2)</pre>
                 stmt;
```



Measuring Input Size

- If input size is longer, the algorithm runs for longer time.
 - Running time depends on input size
 - ▶ Efficiency of an algorithm is computed as a function to which input size is passed as a parameter

3 4/9/20



Problem:

Compiling Telephone directory for mobile users (109 Mobile users in India) in some sorted order.

Assume typical CPU's process up to 108 op/sec (approx. calc)

- Naive sorting algorithm Complexity- n²
 How much time is taken for sorting the directory?
- nlogn sorting algorithm Complexity-nlogn
 How much time is taken for sorting the directory?



Problem:

```
void function(int n)
    int count = 0;
       for (int i=0; i<n; i++)
                for (int j=i; j< i*i; j++)
            if (j%i == 0)
                      for (int k=0; k<j; k++)
                    printf("*");
```

10 4/9/2024

Thank you!