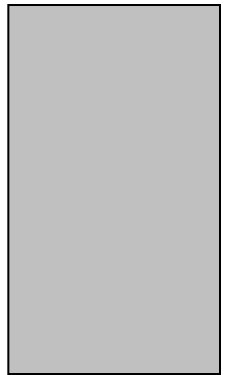


# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```



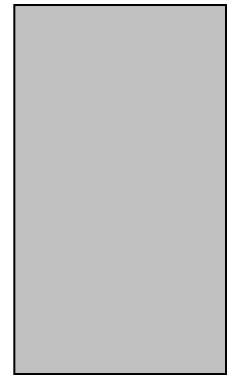
$N = 6$

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```



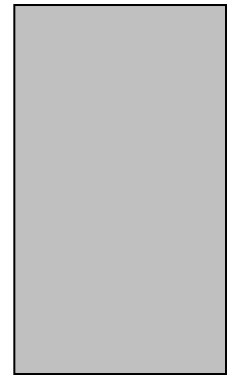
$N = 6$

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

$i$	$v$
0	1



```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$N = 6$

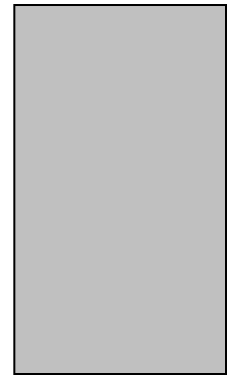
# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true



$N = 6$

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true

```
0 1
```

$N = 6$

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
	2	

```
0 1
```

$N = 6$

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	

```
0 1
```

$N = 6$

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true

```
0 1
```

$N = 6$



# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true

```
0 1
1 2
```

$N = 6$

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
	4	

```
0 1
1 2
```

**N = 6**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	

0	1
1	2

$N = 6$

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true

```
0 1
1 2
```

**N = 6**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true

```
0 1
1 2
2 4
```

**N = 6**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
	8	

0	1
1	2
2	4

**N = 6**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	

0	1
1	2
2	4

**N = 6**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true

0	1
1	2
2	4

**N = 6**



# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true

```
0 1
1 2
2 4
3 8
```

**N = 6**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
	16	

```
0 1
1 2
2 4
3 8
```

**N = 6**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	

```
0 1
1 2
2 4
3 8
```

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true

```
0 1
1 2
2 4
3 8
```

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true

```
0 1
1 2
2 4
3 8
4 16
```

**N = 6**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
	32	

```
0 1
1 2
2 4
3 8
4 16
```

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	

```
0 1
1 2
2 4
3 8
4 16
```

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true

```
0 1
1 2
2 4
3 8
4 16
```

**$N = 6$**



# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true

```
0 1
1 2
2 4
3 8
4 16
5 32
```

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true
	64	

```
0 1
1 2
2 4
3 8
4 16
5 32
```

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true
6	64	

```
0 1
1 2
2 4
3 8
4 16
5 32
```

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true
6	64	true

0	1
1	2
2	4
3	8
4	16
5	32

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true
6	64	true

0	1
1	2
2	4
3	8
4	16
5	32
6	64

**N = 6**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true
6	64	true
	128	

0	1
1	2
2	4
3	8
4	16
5	32
6	64

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true
6	64	true
7	128	

0	1
1	2
2	4
3	8
4	16
5	32
6	64

**$N = 6$**

# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true
6	64	true
7	128	false

0	1
1	2
2	4
3	8
4	16
5	32
6	64

**$N = 6$**



# Powers of Two: Trace

Ex. Print powers of 2 that are  $\leq 2^N$

- Increment  $i$  from 0 to  $N$
- Double  $v$  each time

```
int v = 1;
for (int i = 0; i <= N; i++) {
    System.out.println(i + " " + v);
    v = 2 * v;
}
```

$i$	$v$	$i \leq N$
0	1	true
1	2	true
2	4	true
3	8	true
4	16	true
5	32	true
6	64	true
7	128	false

0	1
1	2
2	4
3	8
4	16
5	32
6	64

**$N = 6$**