

---

**Algorithm** Finding pairs  $(i, j)$  such that  $i \equiv j \pmod x$

---

```

function ModPairs(int[] nums, int x)
     $n \leftarrow \text{length}(\text{nums})$ 
    for  $i \leftarrow 1, n - 1$  do
        for  $j \leftarrow i + 1, n$  do
            if  $i \% j = x$  then
                print("Indices  $(\{i\}, \{j\})$  with values  $\text{nums}[i], \text{nums}[j]$ ")

```

---



---

**Algorithm** Power Set  $\mathcal{P}(\text{int}[])$

---

```

function PowerSet(int[] T)
    Queue<int[]> q                                     ▷ declare queue
    q.queue([])                                         ▷ start with  $\emptyset$ 
    for each  $t \in T$  do                               ▷  $\forall t$ , create new subsets by appending  $t$  to all subsets
        while true do                                  ▷ iterate through queue until  $\emptyset$ 
            int[] subset  $\leftarrow$  q.dequeue()
            int[] newSubset  $\leftarrow$  subset.append(t)      ▷ append  $t$  to subset

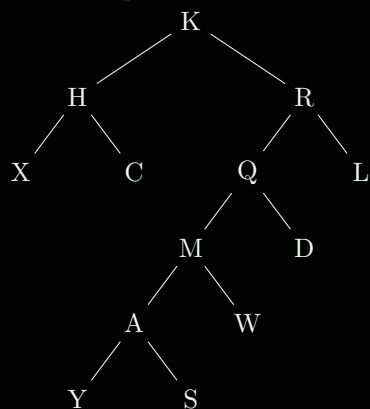
            q.queue(newSubset)                           ▷ queue [subset, t]
            q.queue(subset)                               ▷ requeue subset after
            if subset == [] then
                break                                   ▷ stop at  $\emptyset$ 

    return q

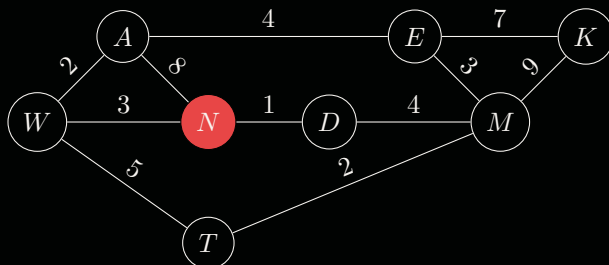
```

---

Tree example



Graph (and array) example



Node	distance
D	$\infty$
W	$\infty$
A	$\infty$
M	$\infty$
T	$\infty$
E	$\infty$
K	$\infty$