How do we know recursion works?

```
//Assume i is a nonnegative integer
//returns 2^i.
public int CalculatesTwoToTheI(int i){
     if(i == 0)
          return 1;
     else
          return 2*CaclulatesTwoToTheI(i-1);
}
```

Why does `CalculatesTwoToTheI(4)` calculate `2^4`?
Convince the people around you!

Making Induction Proofs Pretty

Let $P(n)$ be the predicate " ."
We prove $P(n)$ holds holds for all natural numbers n by induction on n.

Base Case $(\mathrm{n} = 0)$

Inductive Hypothesis:

Inductive Step:

Therefore by the principle of induction.

Making Induction Proofs Pretty

All of our induction proofs will come in 5 easy(?) steps!

1. Define $P(n)$. State that your proof is by induction on n.
2. Show $P(0)$ i.e. show the base case
3. Suppose $P(k)$ for an arbitrary k.
4. Show $P(k+1)$ (i.e. get $P(k) \rightarrow P(k+1)$)
5. Conclude by saying $P(n)$ is true for all n by induction.

More Induction

Induction doesn't **only** work for code!

Show that $\sum_{i=0}^{n} 2^i = 1 + 2 + 4 + \cdots + 2^n = 2^{n+1} - 1$.

Let $P(n) =$"$\sum_{i=0}^{n} 2^i = 2^{n+1} - 1$."

We show $P(n)$ holds for all natural numbers n by induction on n.

Base Case ( )

Inductive Hypothesis:

Inductive Step:

$P(n)$ holds for all $n \geq 0$ by the principle of induction.