$$\left\{\begin{array}{ll} P(1) & \\ P(n) \rightarrow P(n+1) & (n \geq 1) \\ \hline P(n) \quad \forall n \geq 1 & \end{array}\right\}$$ why is this true?

| Prove $P(5)$ | Suppose $P(1)$ and $P(n) \rightarrow P(n+1)$ but also $P(100)$ is false |
|---|---|
| $P(1)$ | $P(99) \rightarrow P(100)$ |
| $P(1) \rightarrow P(2)$ | $\neg P(100)$ |
| $P(2)$ | $\neg P(99)$ |
| $P(2) \rightarrow P(3)$ | $\neg P(98)$ |
| $P(3)$ | $\vdots$ |
| $P(3) \rightarrow P(4)$ | $\neg P(2)$ |
| $P(4)$ | $P(1)$ |
| $P(4) \rightarrow P(5)$ | $P(1) \rightarrow P(2)$ |
| $P(5)$ | $\therefore P(2)$ |
| QED | |

# STRONG INDUCTION

$$\frac{\begin{array}{l} P(1) \\ \big(P(1) \wedge P(2) \wedge P(3) \wedge \dots \wedge P(n)\big) \to P(n+1) \qquad (n \geq 1) \end{array}}{P(n) \quad \forall n \geq 1}$$

theorem: any int $n \geq 2$ can be written as a product of primes.

---

Let $P(n)$ be the prop "$n$ can be written as a product of primes"

$P(2)$ is true

---

prove $\big(P(2) \wedge P(3) \wedge \dots \wedge P(n)\big) \to P(n+1)$

if $n+1$ is prime, done.

else, $n+1 = a \cdot b$, $a, b \geq 2$, $a, b \leq n$

$P(a)$ is true, $P(b)$ is true

$a =$ prod. of primes $\quad b =$ prod. of primes

$(n+1) = a \cdot b =$ (prod of primes)(prod of primes) $=$ prod of primes $\quad \therefore P(n+1)$

prove $n < 2^n \quad \forall n \geq 1$

Let $P(n)$ be the prop "$n < 2^n$"

prove $P(n)\ \forall n \geq 1$. $P(1)$ says "$1 < 2^1$" which is true

prove $P(n) \to P(n+1) \quad (n \geq 1)$

proof

1. $P(n)$ — Hypothesis
2. $n < 2^n$ — Def of $P(n)$, 1
3. $n+1 < 2^n + 1 < 2^n + 2^n = 2 \cdot 2^n = 2^{(n+1)}$ — algebra, prop. of inequalities, $1 < 2^n$
4. $P(n+1)$ — 3 def of $P(n+1)$

$\therefore P(n) \to P(n+1) \quad (n \geq 1)$

$\therefore P(n)\ \forall n \geq 1$ — M.I.

QED

prove $n < 2^n \quad \forall n \geq 1$

Let $P(n)$ be the prop "$n < 2^n$"

prove $P(n)\ \forall n \geq 1$. $P(1)$ says "$1 < 2^1$" which is true

prove $P(n) \to P(n+1) \quad (n \geq 1)$

proof

1. $P(n)$ — Hypothesis
2. $n < 2^n$ — Def of $P(n)$, 1
3. $n+1 < 2^n + 1 < 2^n + 2^n = 2 \cdot 2^n = 2^{(n+1)}$ — algebra, prop. of inequalities, $1 < 2^n$
4. $P(n+1)$ — 3 def of $P(n+1)$

$\therefore P(n) \to P(n+1) \quad (n \geq 1)$

$\therefore P(n)\ \forall n \geq 1$ — M.I.

QED

Prove $3 \mid (n^3 - n)$ $\forall n \geq 0$ using M.I.

Let $P(n)$ be the prop: "$n^3 - n$ is divisible by 3" $(n \geq 0)$

Prove $P(n)$ $\forall n \geq 0$

I Base Case: $P(0)$ says $\underbrace{0^3 - 0}_{0}$ is div. by 3 $\quad 0 = 3 \cdot 0$

$P(0)$ is TRUE

II. Prove $P(n) \to P(n+1)$

Proof:

1. $P(n)$ Hyp
2. $3 \mid (n^3 - n)$ Defn of $P(n)$, 1
3. $3 \mid (n^3 - n + 3n)$ add $3n$ to 2.
4. $3 \mid (n^3 + 2n)$ alg.
5. $3 \mid n^3 + 2n + 3n^2 + n + 1 - (n+1)$ alg.

$3 \mid (n^3 + 3n^2 + 3n + 1) - (n+1)$ alg

Proof:

1. $P(n)$ Hyp

2. $3 \mid (n^3 - n)$ Defn of $P(n)$, 1

3. $3 \mid (n^3 - n + 3n)$ add $3n$ to 2.

4. $3 \mid (n^3 + 2n)$ alg.

5. $3 \mid n^3 + 2n + 3n^2 + n + 1 - (n+1)$ alg.

$3 \mid (n^3 + 3n^2 + 3n + 1) - (n+1)$ alg

$3 \mid (n+1)^3 - (n+1)$ alg

$\therefore P(n+1)$

$\therefore P(n) \Rightarrow P(n+1) \quad \forall n \geq 0$

$\therefore P(n)$ all $\forall n \geq 0$ M.I.

Proof:

1. $P(n)$ Hyp

2. $3 \mid (n^3 - n)$ Def of $P(n)$, 1

3. $3 \mid (n^3 - n + 3n)$ add $3n$ to 2.

4. $3 \mid (n^3 + 2n)$ alg.

5. $3 \mid n^3 + 2n + 3n^2 + n + 1 - (n+1)$ alg.

$3 \mid (n^3 + 3n^2 + 3n + 1) - (n+1)$ alg

$3 \mid (n+1)^3 - (n+1)$ alg

$\therefore P(n+1)$

$\therefore P(n) \Rightarrow P(n+1) \quad \forall n \geq 0$

$\therefore P(n)$ all $\forall n \geq 0$ M.I.

Prove: If $S$ is a finite set, $|\mathcal{P}(S)| = 2^{|S|}$

Let $P(n)$ be the prop: If $|S| = n$, $|\mathcal{P}(S)| = 2^n$ $(n \geq 0)$

$P(0)$ says $|\mathcal{P}(\{\})| = 1$. $\mathcal{P}(\{\}) = \{\emptyset\}$ $P(0)$ is true

Prove $P(n) \to P(n+1)$ $(n \geq 0)$

---

Let $S$ be a set with $n+1$ elements.

Let $s_0$ be some element in $S$.

$S \setminus \{s_0\}$ is a set with $n$ elements

you get $2^n$ subsets of this

Every subset of $S$ is either (a) a subset of this $2^n$

or (b) a subset of this plus $s_0$. $2^n$

$2^{n+1}$

Prove: ~~$AB$~~

$$\overline{A_0 A_1 A_2 \cdots A_n} = \bar{A}_0 + \bar{A}_1 + \bar{A}_2 + \cdots + \bar{A}_n \qquad \forall n \geq 1$$

---

Given: $\overline{AB} = \bar{A} + \bar{B}$

Show $\overline{ABC} = \bar{A} + \bar{B} + \bar{C}$

Prove: ~~AB~~

$$\overline{A_0A_1A_2\cdots A_n} = \bar{A}_0+\bar{A}_1+\bar{A}_2+\cdots+\bar{A}_n \qquad \forall n \geq 1$$

---

Given: $\overline{AB} = \bar{A}+\bar{B}$

Show $\overline{ABC} = \bar{A}+\bar{B}+\bar{C}$

$$\overline{ABC} = \overline{(AB)(BC)} = \overline{(AB)} + \overline{(BC)} = \bar{A}+\bar{B}+\bar{B}+\bar{C} = \bar{A}+\bar{B}+\bar{C}$$

or

$$\overline{ABC} = \overline{(AB)(C)} = \overline{AB} + \bar{C} = \bar{A}+\bar{B}+\bar{C}$$

given this, prove $\overline{ABCD} = \bar{A}+\bar{B}+\bar{C}+\bar{D}$

$$\overline{ABCD} = \overline{(ABC)(D)} = \overline{ABC} + \bar{D} = \bar{A}+\bar{B}+\bar{C}+\bar{D}$$

$$\overline{A_0 A_1 A_2 \cdots A_n A_{n+1}} =$$

$$\overline{(A_0 A_1 A_2 \cdots A_n)(A_{n+1})} =$$

$$\overline{A_0 A_1 A_2 \cdots A_n} + \overline{A_{n+1}} =$$

$$\overline{A_0} + \overline{A_1} + \overline{A_2} + \cdots + \overline{A_n} + \overline{A_{n+1}}$$

# The odd n-pie fight theorem

If $n \geq 3$, someone survives the fight

RULES

$n$ people

$n$ pies

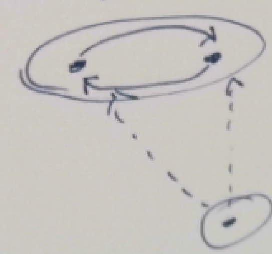
The distance between any 2 people is unique

Each person throws their pie at the person closest to them.

Let $P(n)$ be the prop: In an $n$-pie fight, someone survives.

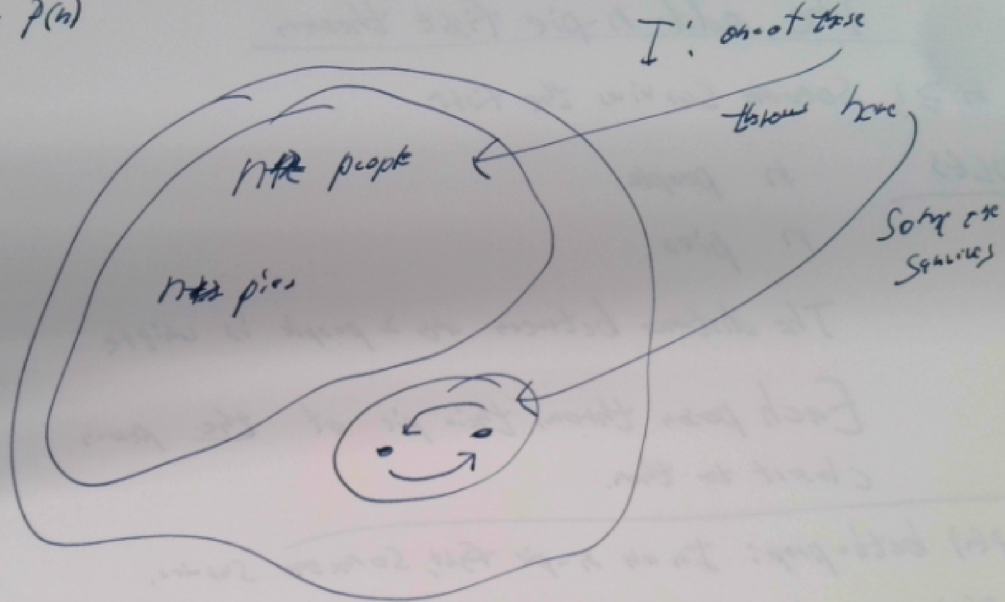
Prove $P(n) \to P(n+2)$ and $P(3)$

BASE CASE: $P(3)$



TRUE

Assume $P(n)$




II By $P(n)$, someone survives.

$\therefore P(n) \rightarrow P(n+1)$

Postage Stamp Problem

Given an unlimited supply of 5¢ & 7¢ stamps,

can you make exact postage for any amount $\geq 28$?

Let $P(n)$ be the prop: you can make $n$¢ from a bunch of
5¢ & 7¢

Prove $P(n)$ $n \geq 28$

Base case: $P(28)$ $28 = 7+7+7+7$ $\therefore P(28)$ is true

---

Prove $P(n) \rightarrow P(n+1)$ $n \geq 28$

1. $P(n)$ hyp

2. If the solution for $P(n)$ involves at least $7+7$,
replace those with $5+5+5$

Otherwise: at most one 7¢ stamp; the other stamps are each 5¢
and add up to at least 21 $\therefore$ we have at least 4 5¢ stamps
Remove 4 5¢, add 3 7¢.

$\therefore P(n+1)$ etc.

Wed 06 Nov 2024, 13:30:36 ~ : screengrabber — Konsole (nick) linux.engrcs.com — K...

(nick) linux.engrcs.com — Konsole

File Edit View Bookmarks Settings Help

```c
// Recursive stamp problem solver - njm
// 5 and 7 cent stamps, assumes total is at least 28

void solve(int total,int *five, int *seven)
{
// base case
  if (total==28){ // 7+7+7+7
    *five=0;*seven=4;return;
  }

// solve for total-1
  solve(total-1,five,seven);

// If there are *at least* two 7-cent stamps,
// we can replace them with three 5's
// (total postage increases by 1).

  if (*seven >= 2){
    (*seven)=(*seven)-2;  // remove two 7 cent stamps  (-14)
    (*five)+=3;          // and add 3 five cent stamps (+15)
    return;             // net change is +1 cent :)
  }

// Otherwise, there's only one 7 cent stamp,
// so there must be at least four 5 cent stamps.
// Replace four 5 cent stamps with three 7 scent stamps
// (total postage increases by 1).

  (*five)=(*five)-4;    // remove four 5 cent stamps (-20)
  (*seven)=(*seven)+3; // and add three 7 cent stamps (+21)
"solve.c" 32L, 941B
```

Wed 06 Nov 2024, 13:30:42 ~ : screengrabber — Konsole (nick) linux.engrcs.com — K...

(nick) linux.engrcs.com — Konsole

File Edit View Bookmarks Settings Help

```
// Recursive stamp problem solver - njm
// 5 and 7 cent stamps, assumes total is at least 28

void solve(int total,int *five, int *seven)
{
// base case
  if (total==28){ // 7+7+7+7
    *five=0;*seven=4;return;
  }

// solve for total-1
  solve(total-1,five,seven);

// If there are *at least* two 7-cent stamps,
// we can replace them with three 5's
// (total postage increases by 1).

  if (*seven >= 2){
    (*seven)=(*seven)-2;  // remove two 7 cent stamps  (-14)
    (*five)+=3;          // and add 3 five cent stamps (+15)
    return;             // net change is +1 cent :)
  }

// Otherwise, there's only one 7 cent stamp,
// so there must be at least four 5 cent stamps.
// Replace four 5 cent stamps with three 7 scent stamps
// (total postage increases by 1).

  (*five)=(*five)-4;    // remove four 5 cent stamps (-20)
  (*seven)=(*seven)+3; // and add three 7 cent stamps (+21)
"solve.c" 32L, 941B
```

Wed 06 Nov 2024, 13:32:00 ~ : screengrabber — Konsole (nick) linux.engrcs.com — K...

(nick) linux.engrcs.com — Konsole

File Edit View Bookmarks Settings Help

```
void solve(int total,int *five, int *seven)
{
// base case
  if (total==28){ // 7+7+7+7
    *five=0;*seven=4;return;
  }

// solve for total-1
  solve(total-1,five,seven);

// If there are *at least* two 7-cent stamps,
// we can replace them with three 5's
// (total postage increases by 1).

  if (*seven >= 2){
    (*seven)=(*seven)-2;  // remove two 7 cent stamps  (-14)
    (*five)+=3;          // and add 3 five cent stamps (+15)
    return;             // net change is +1 cent :)
  }

// Otherwise, there's only one 7 cent stamp,
// so there must be at least four 5 cent stamps.
// Replace four 5 cent stamps with three 7 scent stamps
// (total postage increases by 1).

  (*five)=(*five)-4;    // remove four 5 cent stamps (-20)
  (*seven)=(*seven)+3; // and add three 7 cent stamps (+21)
  return;             // next change is +1 cent
}
```

Wed 06 Nov 2024, 13:33:18 ~ : screengrabber — Konsole (nick) linux.engrcs.com — K...

(nick) linux.engrcs.com — Konsole

File Edit View Bookmarks Settings Help

```c
#include <stdio.h>
// 5 and 7 cent stamps, at least 28 cents postage
// njm

void solve(int, int*, int*); // this is the solver

// postage is given on command line
void main(int argc, char **argv)
{
  int total,five,seven;

// legal argument?
  if ((argc != 2) || (sscanf(argv[1],"%d",&total) != 1) || (total < 28)){
    printf("Usage: %s postage (at least 28)\n",argv[0]);
    return;
  }

// solve it!
  solve(total,&five,&seven);

// and display results
  printf("\n");
  for (int i=0;i<five;i++) printf("5 "); // show the stamps :)
  for (int i=0;i<seven;i++) printf("7 ");
  printf("\n\n");
  printf("# of 5 cent stamps: %d\n# of 7 cent stamps: %d\n",
           five,seven);
}
~
~
"stamp.c" 28L, 684B
```

Wed 06 Nov 2024, 13:34:45 ~ : screengrabber — Konsole (nick) linux.engrcs.com — K...

(nick) linux.engrcs.com — Konsole

File Edit View Bookmarks Settings Help

```
{2204} stamp 28

7 7 7 7

# of 5 cent stamps: 0
# of 7 cent stamps: 4
{2205} stamp 289

5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7

# of 5 cent stamps: 55
# of 7 cent stamps: 2
{2206} stamp 29

5 5 5 7 7

# of 5 cent stamps: 3
# of 7 cent stamps: 2
{2207} stamp 30

5 5 5 5 5 5

# of 5 cent stamps: 6
# of 7 cent stamps: 0
{2208} stamp 31

5 5 7 7 7

# of 5 cent stamps: 2
# of 7 cent stamps: 3
{2209}
```

Wed 06 Nov 2024, 13:36:16 ~ : screengrabber — Konsole (nick) linux.engrcs.com — K...

(nick) linux.engrcs.com — Konsole

File Edit View Bookmarks Settings Help

```
5 5 5 5 5 5

# of 5 cent stamps: 6
# of 7 cent stamps: 0
{2208} stamp 31

5 5 7 7 7

# of 5 cent stamps: 2
# of 7 cent stamps: 3
{2209} stamp 32

5 5 5 5 5 7

# of 5 cent stamps: 5
# of 7 cent stamps: 1
{2210} stamp 33

5 7 7 7 7

# of 5 cent stamps: 1
# of 7 cent stamps: 4
{2211} stamp 34

5 5 5 5 7 7

# of 5 cent stamps: 4
# of 7 cent stamps: 2
{2212} stamp 34377621
Segmentation fault
{2213}
```

$$n! = 1\cdot 2\cdot 3\cdots(n-1)(n)$$

$$3! = 1\cdot 2\cdot 3 = 6$$

$$4! = \underbrace{(1\cdot 2\cdot 3)}4 = 24$$

$$4! = 3!\cdot 4$$

$$n! = (n-1)!\cdot n$$

$$(n+1)! = (n!)(n+1) \qquad n \geq 1$$

$$2 \qquad \text{if } n=1$$

Wed 06 Nov 2024, 13:42:10 ~ : screengrabber — Konsole (nick) linux.engrcs.com — K...

(nick) linux.engrcs.com — Konsole

File Edit View Bookmarks Settings Help

```c
#include <stdio.h>
int fact(int);

int main()
{
  int n;
  char temp[120];

  printf("Enter n: ");
  fgets(temp,120,stdin);
  sscanf(temp,"%d",&n);

// calculate n!
  printf("%d!=%d\n",n,fact(n));
}

int fact(int n)
{
  if (n==0) return(1);
  return(n*fact(n-1));
}
~
~
~
~
~
~
~
~
~
"fact.c" 21L, 266B
```

Wed 06 Nov 2024, 13:42:56 ~ : screengrabber — Konsole (nick) linux.engrcs.com — K...

(nick) linux.engrcs.com — Konsole

File Edit View Bookmarks Settings Help

```c
#include <stdio.h>
int fact(int);

int main()
{
  int n;
  char temp[120];

  printf("Enter n: ");
  fgets(temp,120,stdin);
  sscanf(temp,"%d",&n);

// calculate n!
  printf("%d!=%d\n",n,fact(n));
}

int fact(int n)
{
  if (n==0) return(1);
  return(n*fact(n-1));
}
~
~
~
~
~
~
~
~
~
"fact.c" 21L, 266B
```

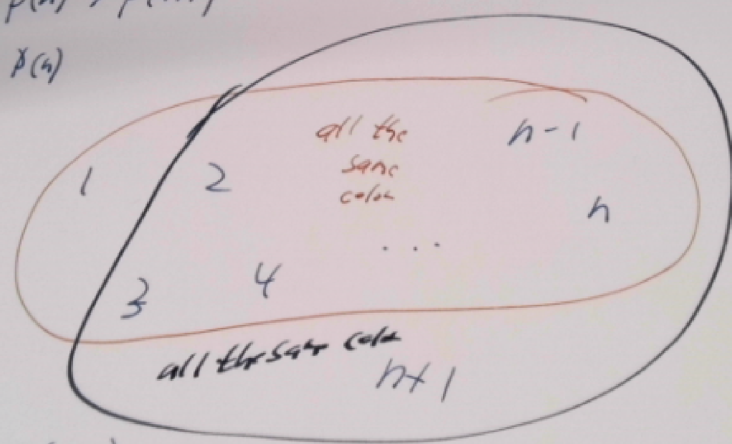
Prove: all horses are ~~brown~~ the same color

Let $P(n)$ be the prop: "in any group of $n$ horses, all of them are ~~brown~~ the same color"

$P(1)$ True

Prove $P(n) \rightarrow P(n+1)$

given $P(n)$




$\therefore P(n+1)$

$\therefore P(n)\ \forall n \geq 1$