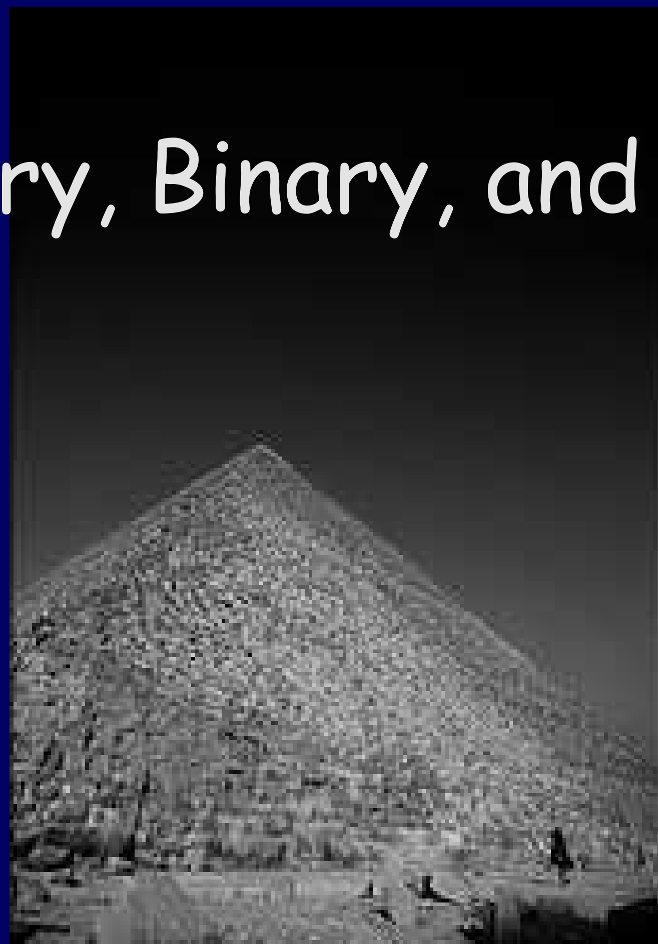
Great Theoretical Ideas In Computer Science

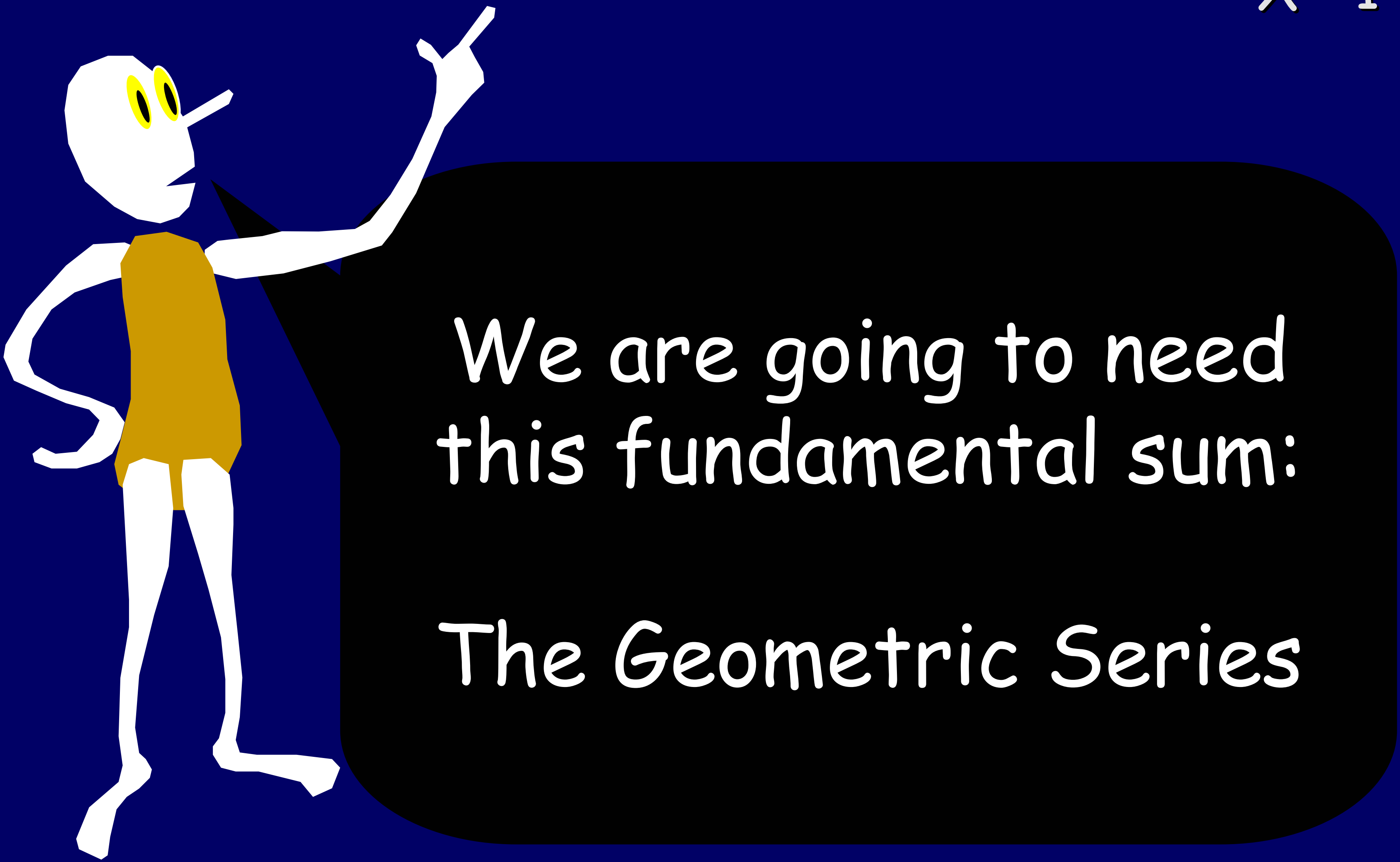
Steven Rudich CS 15-251 Spring 2004

Lecture 3 Jan 20, 2004 Carnegie Mellon University

Unary, Binary, and Beyond



$$1 + X^1 + X^2 + X^3 + \ldots + X^{n-2} + X^{n-1} = \frac{X^n - 1}{X - 1}$$



A Frequently Arising Calculation

$$(X-1)\,(1 + X^1 + X^2 + X^3 + \ldots + X^{n-2} + X^{n-1})$$

$$= X^1 + X^2 + X^3 + \ldots + X^{n-1} + X^n$$
$$- 1 - X^1 - X^2 - X^3 - \ldots - X^{n-2} - X^{n-1}$$

$$= -1 + X^n$$

$$= X^n - 1$$

The Geometric Series

$$(X-1)\,(1 + X^1 + X^2 + X^3 + \ldots + X^{n-1}) = X^n - 1$$

$$1 + X^1 + X^2 + X^3 + \ldots + X^{n-2} + X^{n-1} = \frac{X^n - 1}{X - 1}$$

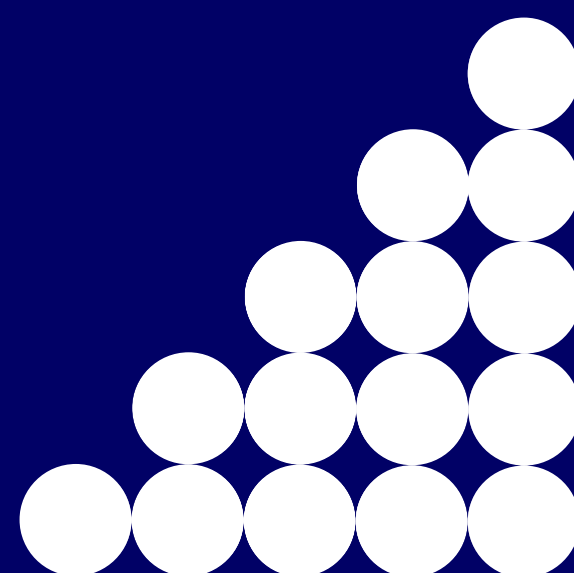
when $X \neq 1$


Last time we talked about unary notation.

n^{th} Triangular Number

$$\Delta_n = 1 + 2 + 3 + \ldots + n\text{-}1 + n$$

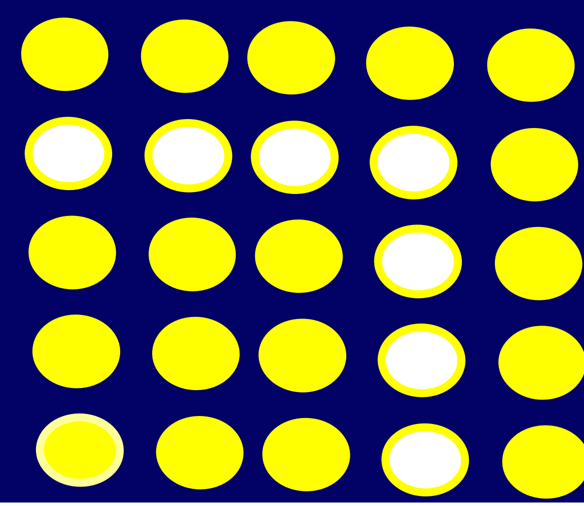
$$= n(n+1)/2$$



n^{th} Square Number

$$n = 1 + 3 + \ldots + 2n-1$$

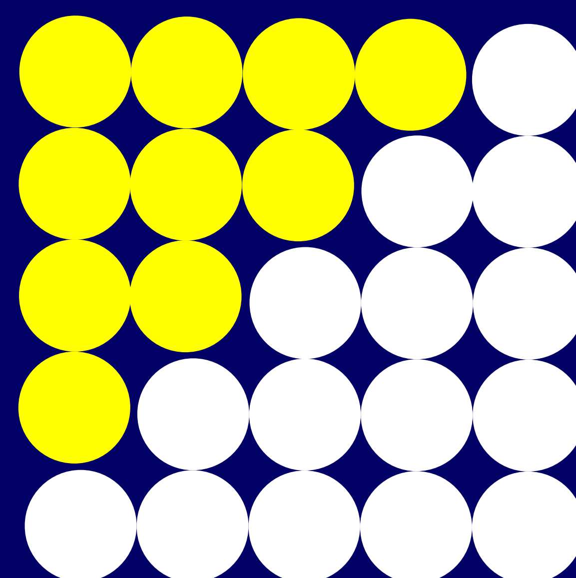
= Sum of first n odd numbers



n^{th} Square Number

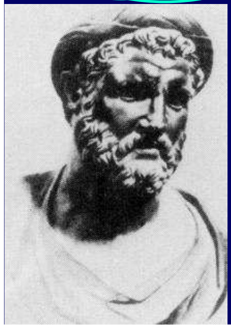
$$n = \Delta_n + \Delta_{n-1}$$

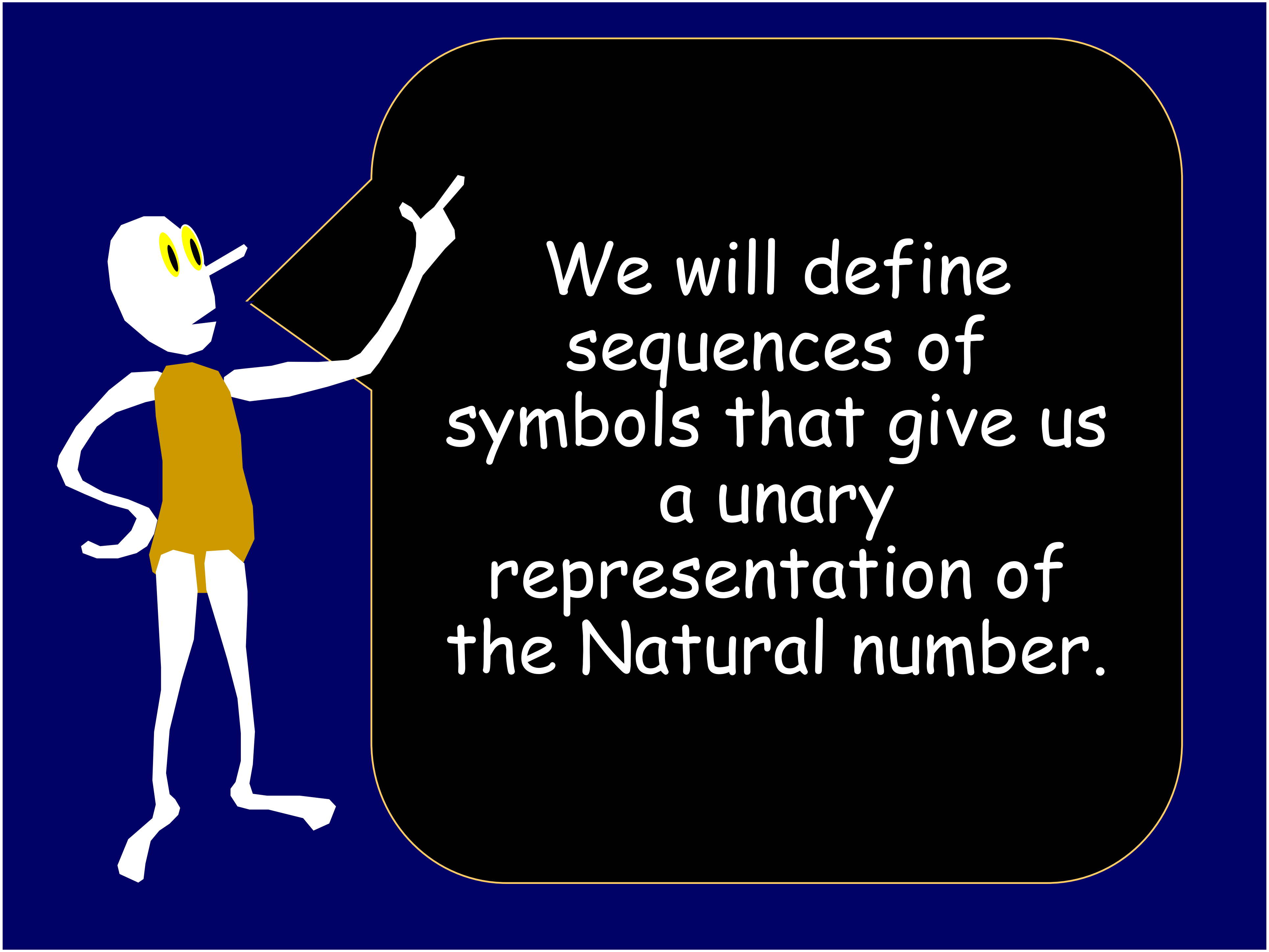
$$= n^2$$

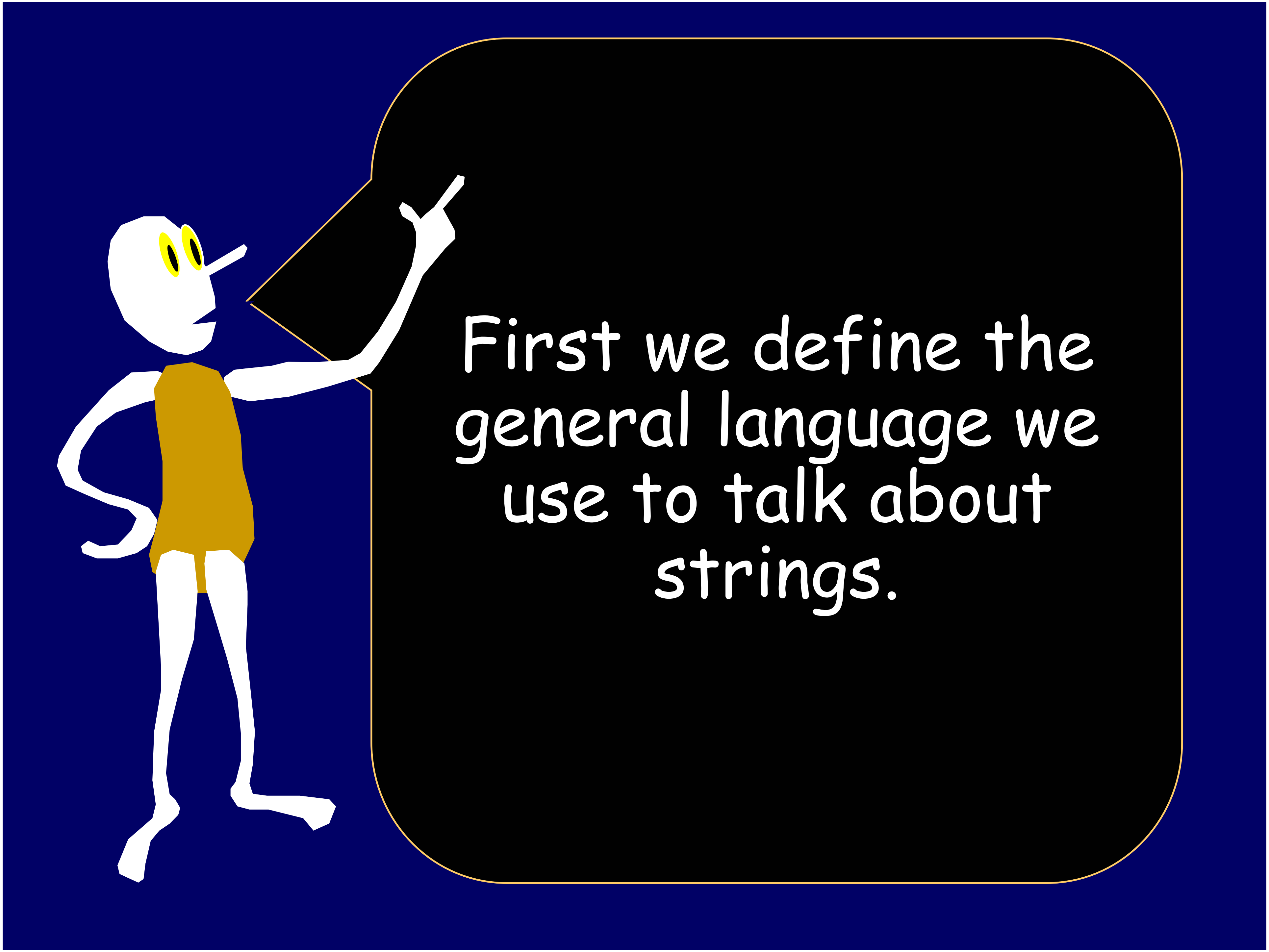


$$(\Delta_n)^2 = (\Delta_{n-1})^2 + \square_n$$

$$(\Delta_n)^2 = \square + \square + \ldots + \square_n$$




We will define sequences of symbols that give us a unary representation of the Natural number.


First we define the general language we use to talk about strings.

Strings Of Symbols.

We take the idea of symbol and sequence of symbols as primitive.

Let Σ be any fixed finite set of symbols. Σ is called an alphabet, or a set of symbols.

Examples:

$\Sigma = \{0,1,2,3,4\}$

$\Sigma = \{a,b,c,d, \ldots, z\}$

Σ = all typewriter symbols.

Strings over the alphabet Σ.

A string is a sequence of symbols from Σ. Let s and t be strings. Let st denote the concatenation of s and t, i.e., the string obtained by the string s followed by the string t.

Define Σ^+ by the following inductive rules:

$$x \in \Sigma \Rightarrow x \in \Sigma^+$$

$$s,t \in \Sigma^+ \Rightarrow st \in \Sigma^+$$

Intuitively, Σ^+ is the set of all finite strings that we can make using (at least one) letters from Σ.

Σ^*

Define ε be the empty string. I.e.,
$X\varepsilon Y = XY$ for all strings X and Y.
ε is also called the string of length 0.

Define $\Sigma^0 = \{ \varepsilon \}$

Define $\Sigma^* = \Sigma^+ \cup \{\varepsilon\}$

Intuitively, Σ^* is the set of all finite strings that we can make using letters from Σ, including the empty string.

The Natural Numbers

N = { 0, 1, 2, 3, . . .}

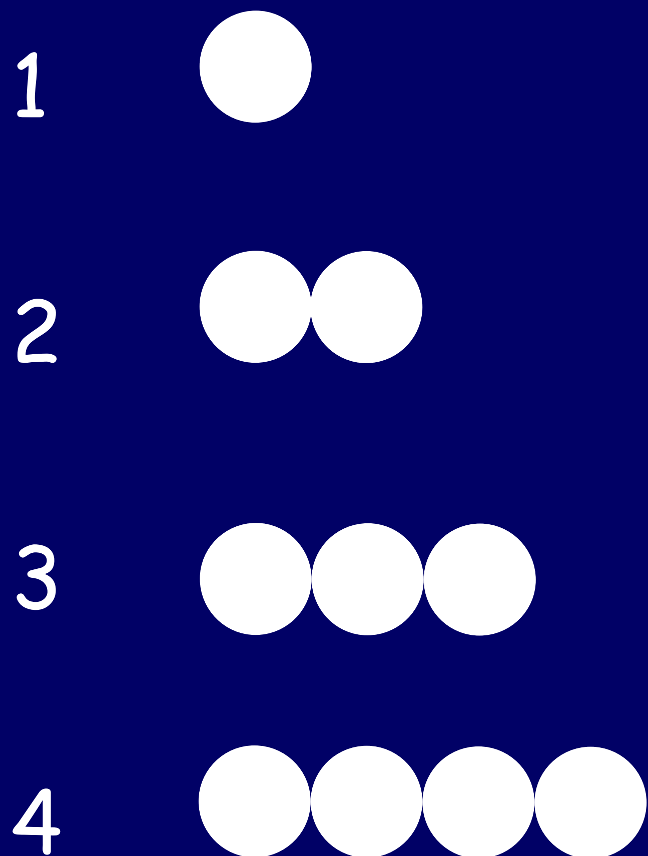
Notice that we include 0 as a Natural number.

"God Made The Natural Numbers. Everything Else Is The Work Of Man."

Kronecker

$N = \{ 0, 1, 2, 3, . . . \}$

Last Time: Unary Notation



To handle the notation for zero, we introduce a small variation on unary.

Peano Unary (PU)

0	0
1	S0
2	SS0
3	SSS0
4	SSSS0
5	SSSSS0
6	SSSSSS0

Giuseppe Peano [1889]

Each number is a sequence of symbols in $\{S, 0\}^+$

0	0
1	S0
2	SS0
3	SSS0
4	SSSS0
5	SSSSS0
6	SSSSSS0

Peano Unary Representation of Natural Number

$\mathrm{N} = \{ 0, S0, SS0, SSS0, \ldots \}$

0 is a n<u>atural number</u> called <u>zero</u>.

Set notation: $0 \in \mathrm{N}$

If X is a natural number, then SX is a <u>natural number</u> called <u>successor of X</u>.

Set notation: $X \in \mathrm{N} \Rightarrow SX \in \mathrm{N}$

Inductive Definition of +

$\mathrm{N} = \{ 0, S0, SS0, SSS0, \ldots \}$

Inductive definition of addition (+):

$X, Y \in \mathrm{N} \Rightarrow$

X "+" $0 = X$

X "+" $SY = S(X$"+"$Y)$

$S0 + S0 = SS0$ (i.e., "1+1=2")

Proof:

$S0 + S0 =$
$S(S0 + 0) = S(S0) =$
$SS0$

$X, Y \in \mathbb{N} \Rightarrow$
$X \text{ "+" } 0 = X$
$X \text{ "+" } SY = S(X \text{"+"} Y)$

Inductive Definition of *

N = { 0, S0, SS0, SSS0, . . .}

Inductive definition of <u>times (*):</u>

$X, Y \in N \Rightarrow$

X "*" 0 = 0

X "*" SY = (X"*"Y) + X

Inductive Definition of ^

$\mathbb{N}$ = { 0, S0, SS0, SSS0, . . .}

Inductive definition of raised to the (^):

X, Y $\in \mathbb{N} \Rightarrow$

X "^" 0 = 1 [or $X^0 = 1$]

X "^" SY = (X"^"Y) * X [or $X^{SY} = X^Y * X$]

$N = \{ 0, S0, SS0, SSS0, \ldots\}$

Defining < for N:

$\forall x,y \in N$
"x > y" is TRUE $\Leftrightarrow$ "y < x" is FALSE
"x > y" is TRUE $\Leftrightarrow$ "y > x" is FALSE

"x+1 > 0" is TRUE
"x+1 > y+1" is TRUE $\Rightarrow$ "x > y" is TRUE

$$N = \{\, 0, 1, 2, 3, \ldots \}$$

Defining partial minus for N:

$\forall\, x,y \in N$

$x - 0 = x$

$x > y \Rightarrow$

$$(x+1) - (y+1) = x - y$$

$$a = [a \text{ DIV } b]*b + [a \text{ MOD } b]$$

Defining DIV and MOD for $\mathbb{N}$:

$\forall$ a,b $\in \mathbb{N}$

a<b $\Rightarrow$

a DIV b = 0

a$\geq$b>0 $\Rightarrow$

a DIV b = 1 + (a-b) DIV b

The maximum number of times b goes into a without going over.

a MOD b = a – [b*(a DIV b)]

The remainder when a is divided by b.

$$45 = [45 \text{ DIV } 10]*10 + [45 \text{ MOD } 10]$$
$$= 4*10 + 5$$

Defining DIV and MOD for $\mathbb{N}$:

$\forall\, a,b \in \mathbb{N}$

$a<b \Rightarrow$

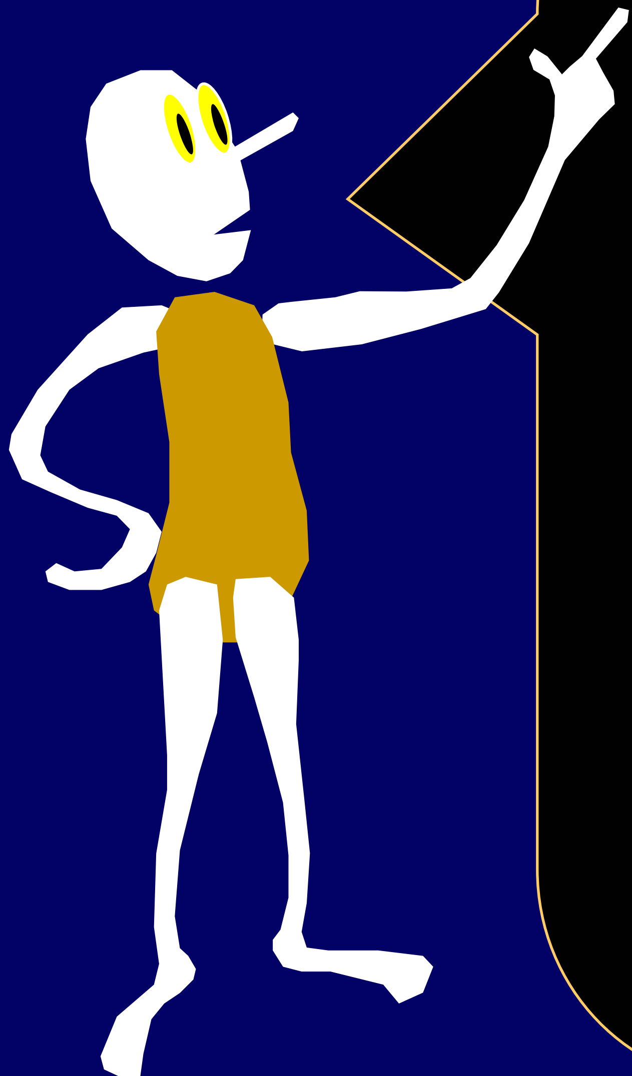
$a \text{ DIV } b = 0$

$a \geq b > 0 \Rightarrow$

$a \text{ DIV } b = 1 + (a-b) \text{ DIV } b$

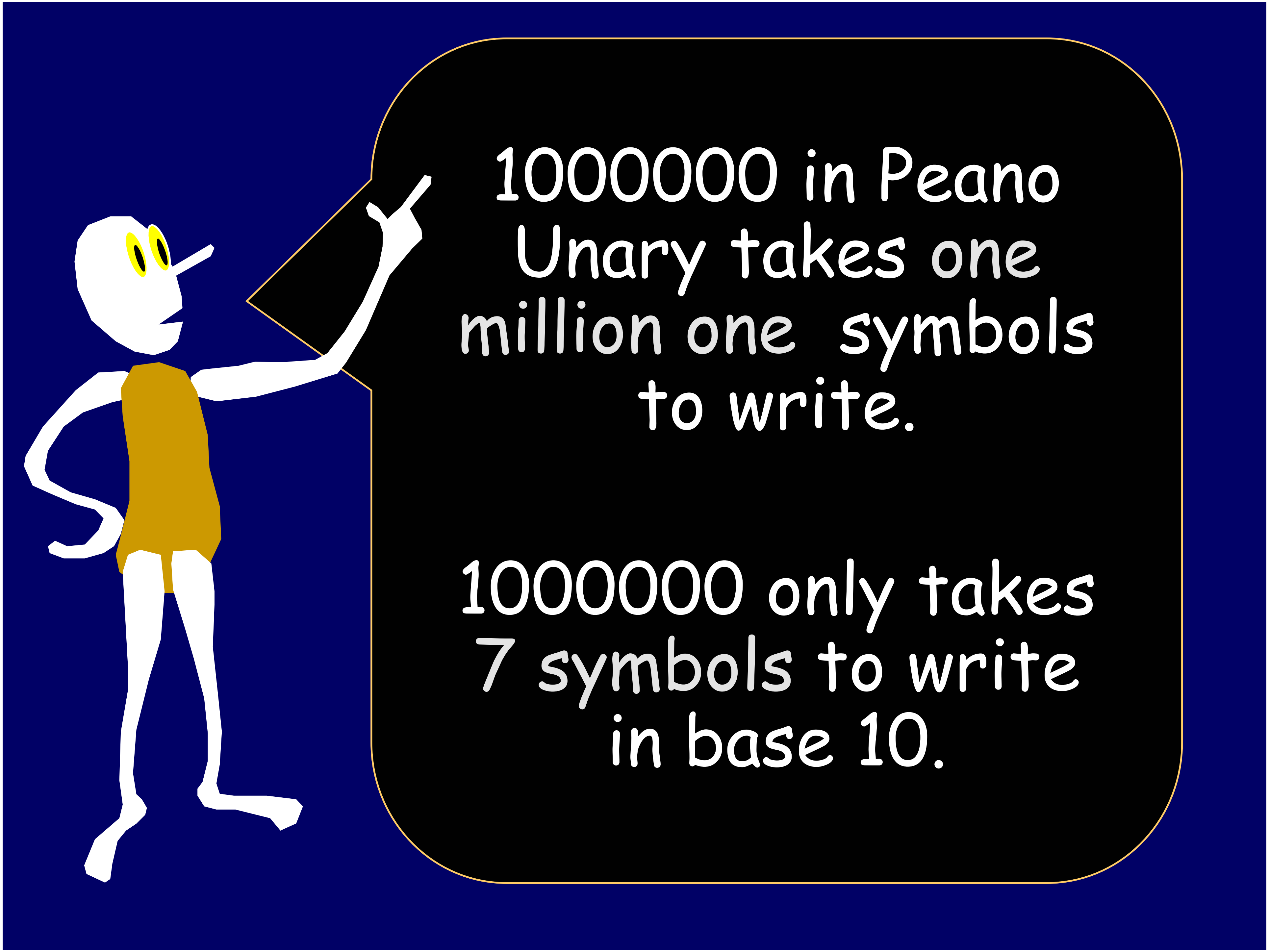
$a \text{ MOD } b = a - [b*(a \text{ DIV } b)]$

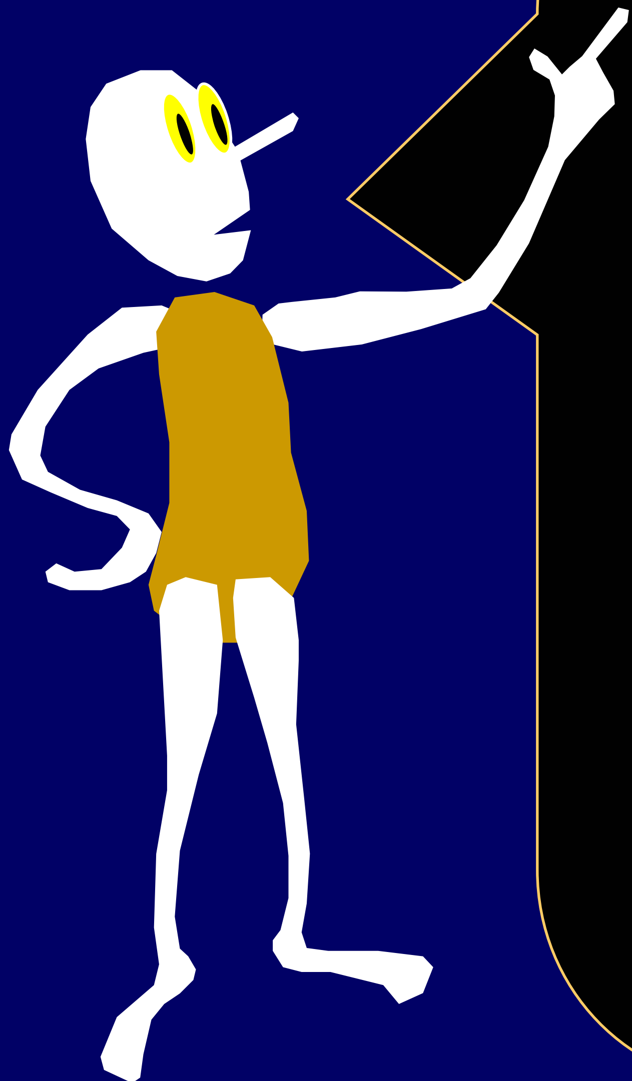
We have defined the Peano representation of the Naturals, with a notion of +, *, ^, <, -, DIV, and MOD.



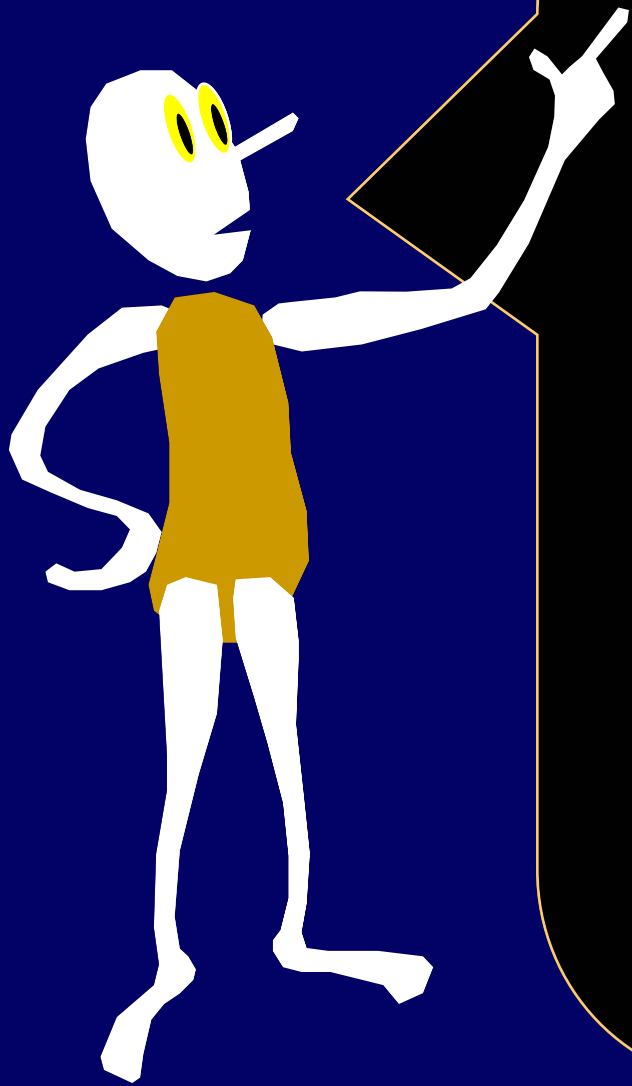
PU takes size n+1 to represent the number n.

Higher bases are much more compact.


1000000 in Peano Unary takes one million one symbols to write.
1000000 only takes 7 symbols to write in base 10.

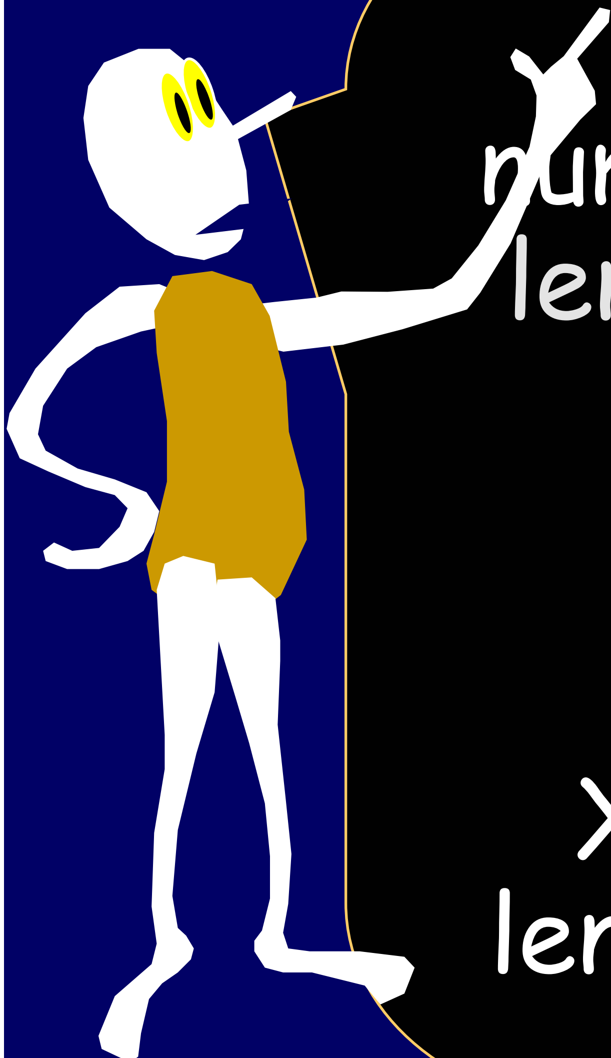


Let’s define base 10 representation of PU numbers.



Let Σ = {0, 1,2,3,4,5,6,7,8,9} be our symbol alphabet.

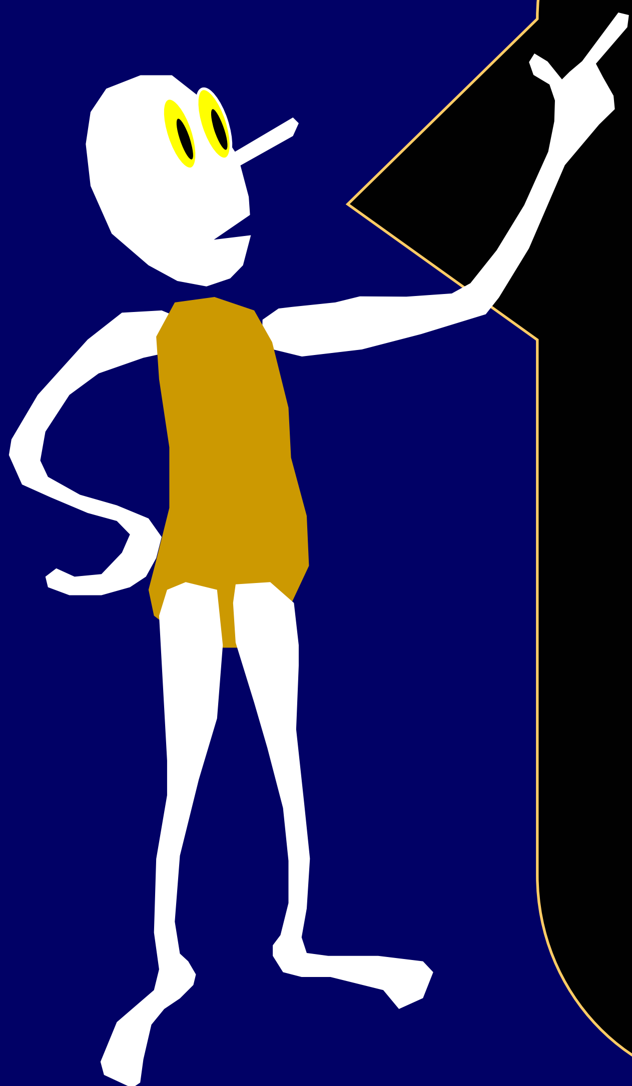
Any string in Σ^+ will be called a decimal number.



Let X be a decimal number. Let's define the length of X inductively:

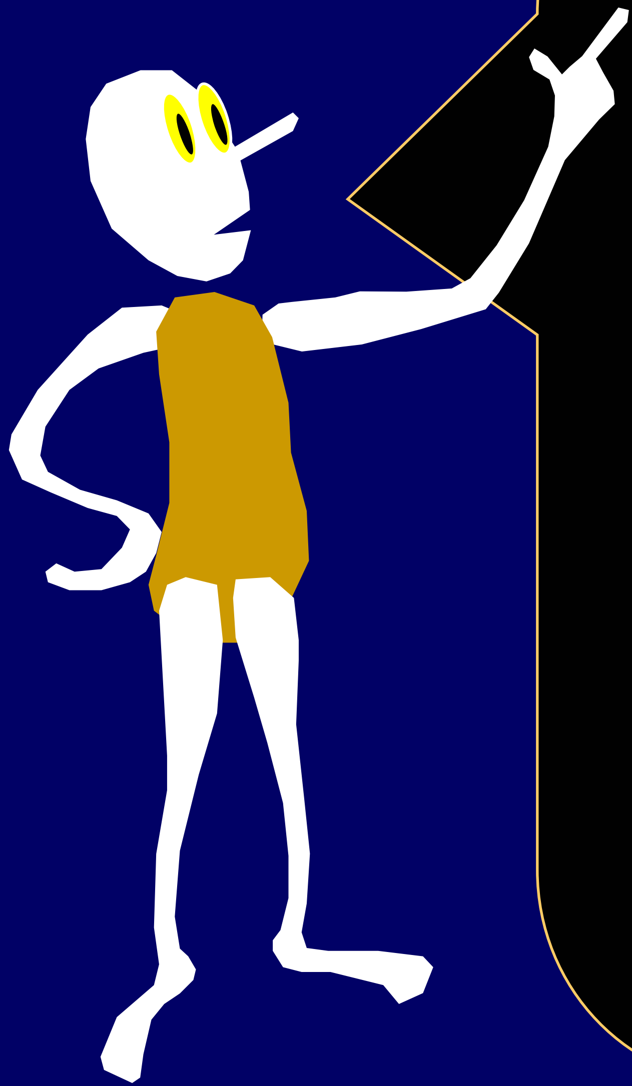
$$\text{length}(\varepsilon) = 0$$

$$X = aY,\ a \in \Sigma,\ Y \in \Sigma^* \Rightarrow \text{length}(X) = S(\text{length}(Y))$$



Let X be decimal.
Let n (unary) =
length(X).

For each unary $i<n$,
we want to be able
to talk about the i^{th}
symbol of X.



The i^{th} symbol of X.

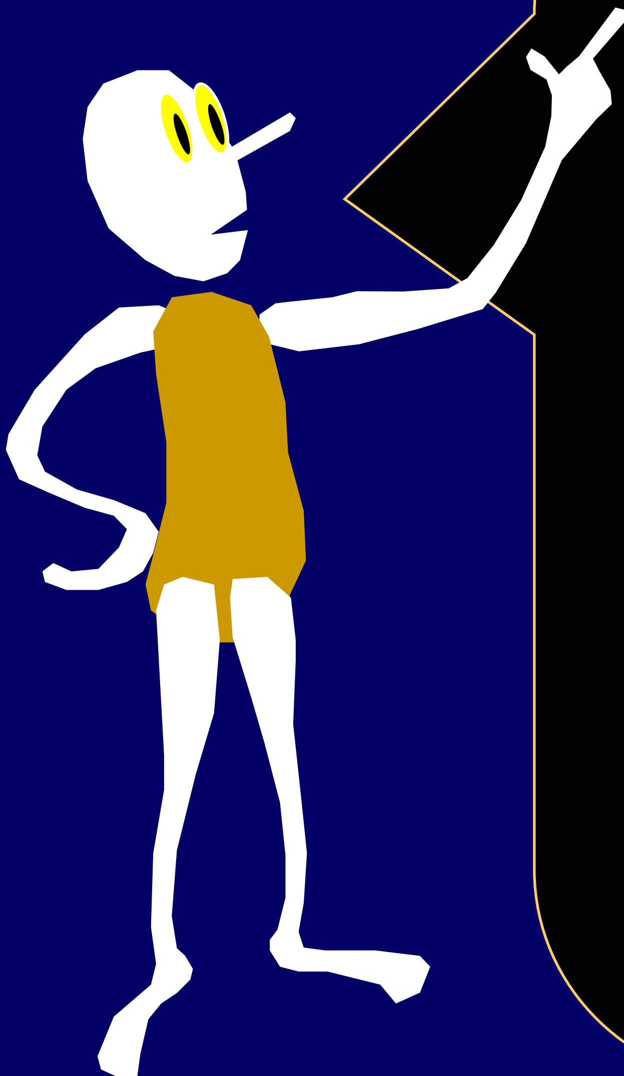
Defining rule:

X = PaS where

P,S$\in \Sigma^*$ and a$\in\Sigma$.

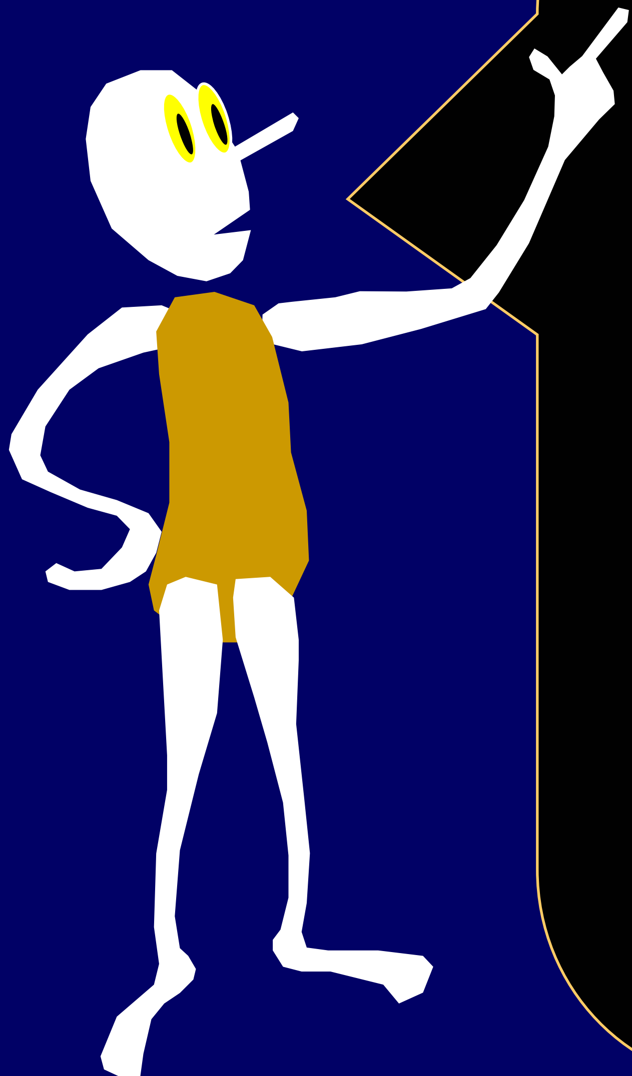
i=length(S).

$\Rightarrow$ a is the i^{th} symbol of X.



For any string X, we can define its length $n \in PU$. For all $i \in PU$ s.t. $i<n$, we can define the i^{th} symbol a_i.

$$X = a_{n-1}\, a_{n-2} \ldots a_0$$



We define a base conversion function, $\mathrm{Base}_{10 \text{ to } 1}(X)$, to convert any decimal X to its unary representation.

Initial Cases, length(X)=1

$\text{Base}_{10 \text{ to } 1}(0) = 0$

$\text{Base}_{10 \text{ to } 1}(1) = S0$

$\text{Base}_{10 \text{ to } 1}(2) = SS0$

$\text{Base}_{10 \text{ to } 1}(3) = SSS0$

……

$\text{Base}_{10 \text{ to } 1}(9) = SSSSSSSSS0$

Suppose n=length(X)>S0

For all $i<n$, let a_i be the i^{th} symbol of X.
Hence, $X = a_{n-1}\ a_{n-2}\ \ldots a_0$.

Define $Base_{10\text{ to }1}(X) =$

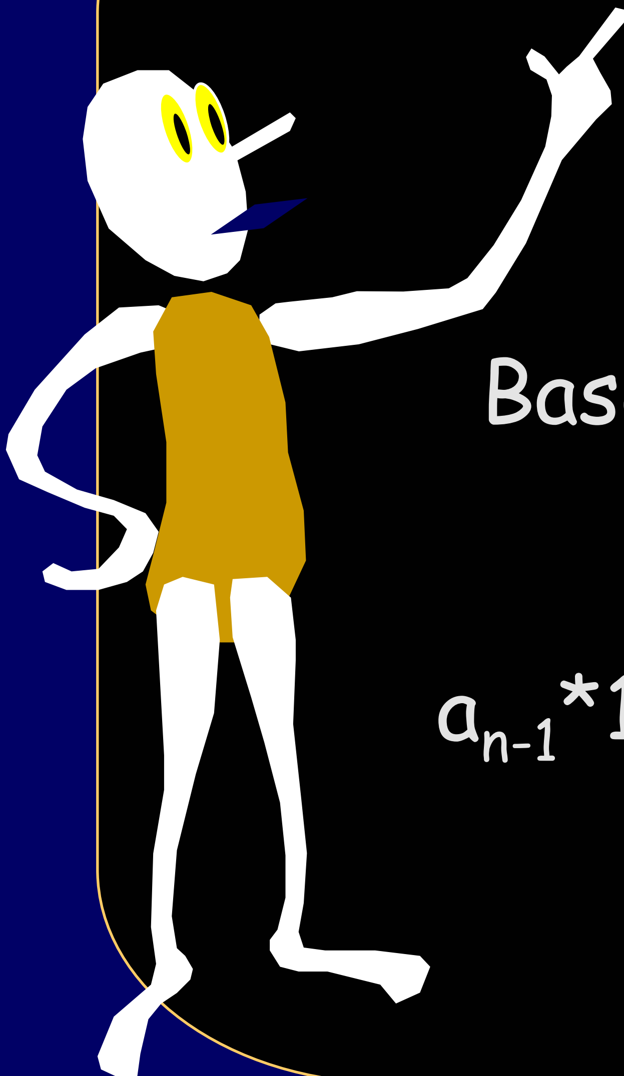
$$\Sigma_{i<n}\ Base_{10\text{ to }1}(a_i)*10^i$$

where +, *, and ^ are defined over PU

Example X= 238

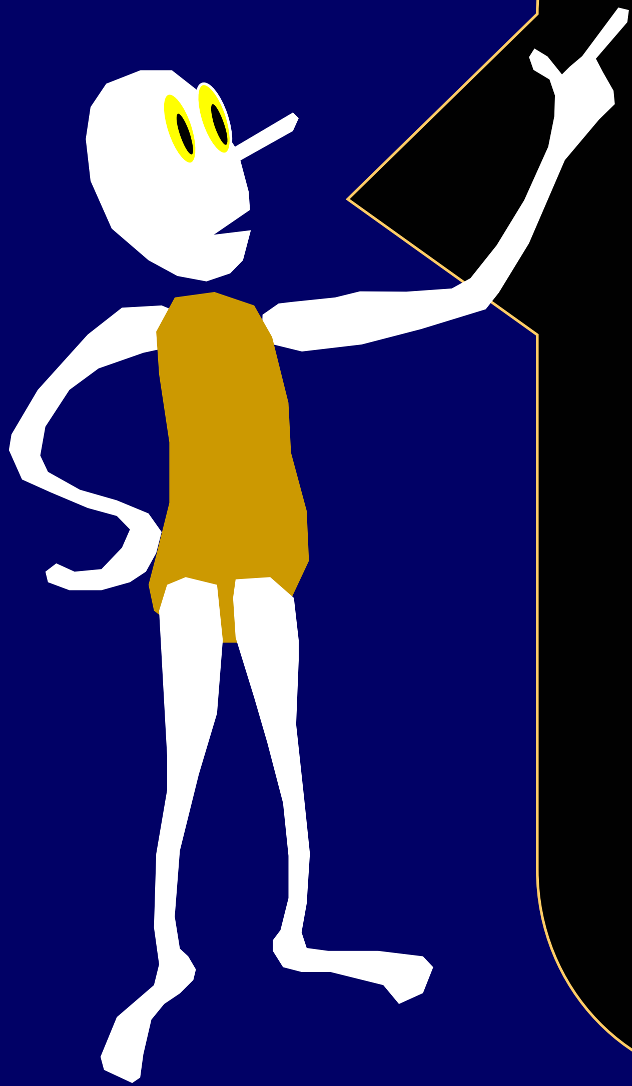
$Base_{10\ to\ 1}(238) =$

$2*100 + 3*10 + 8$



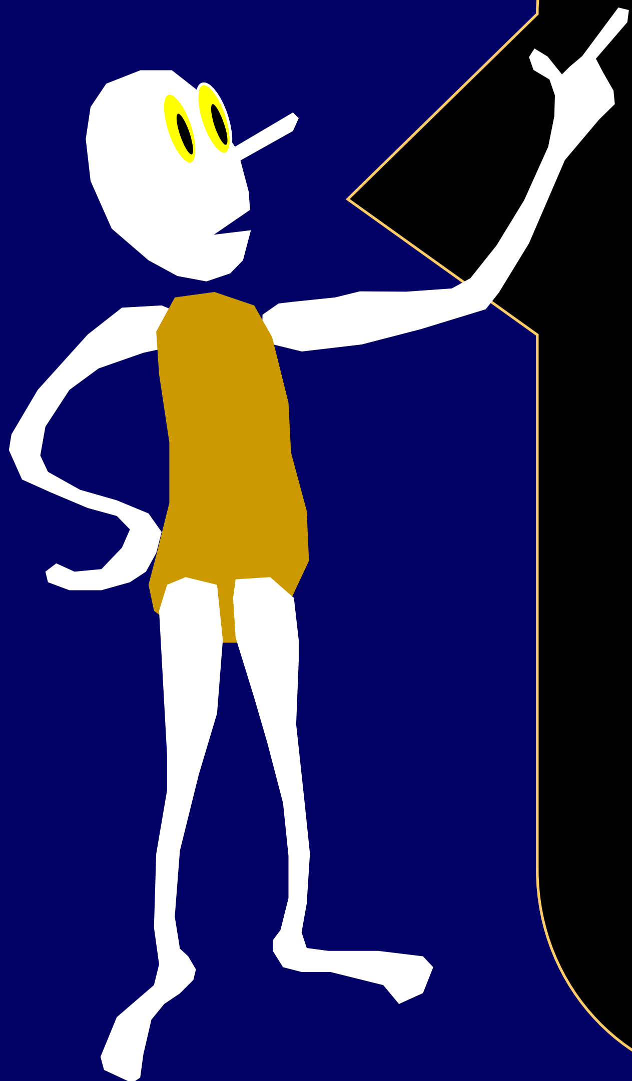
$$\text{Base}_{10 \text{ to } 1}\,(a_{n-1}\, a_{n-2} \ldots a_0) =$$

$$a_{n-1} * 10^{n-1} + a_{n-2} * 10^{n-2} + \ldots + a_0\, 10^0$$



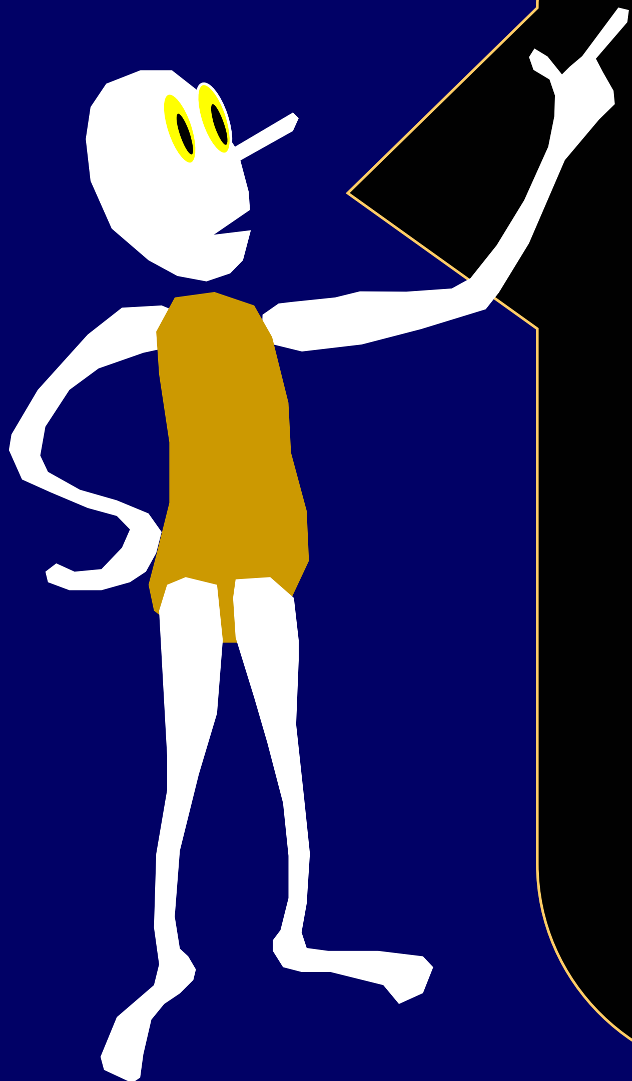
Now we want to go back the other way.

We want to convert PU to decimal.



No Leading Zero:

Let NLZ be the set of decimal numbers with no leading 0 (leftmost symbol $\neq$ 0), or the decimal number 0.



We define Base$_{1 \text{ to } 10}$ from PU to NLZ

It will turn out to be the inverse function to Base$_{10 \text{ to } 1}$.

One digit cases.

$Base_{1\ to\ 10}(0)$ = "0"

$Base_{1\ to\ 10}(S0)$ = "1"

$Base_{1\ to\ 10}(SSo)$ ="2"

….

$Base_{1\ to\ 10}(SSSSSSSSS0)$ = "9"

Suppose X > 9

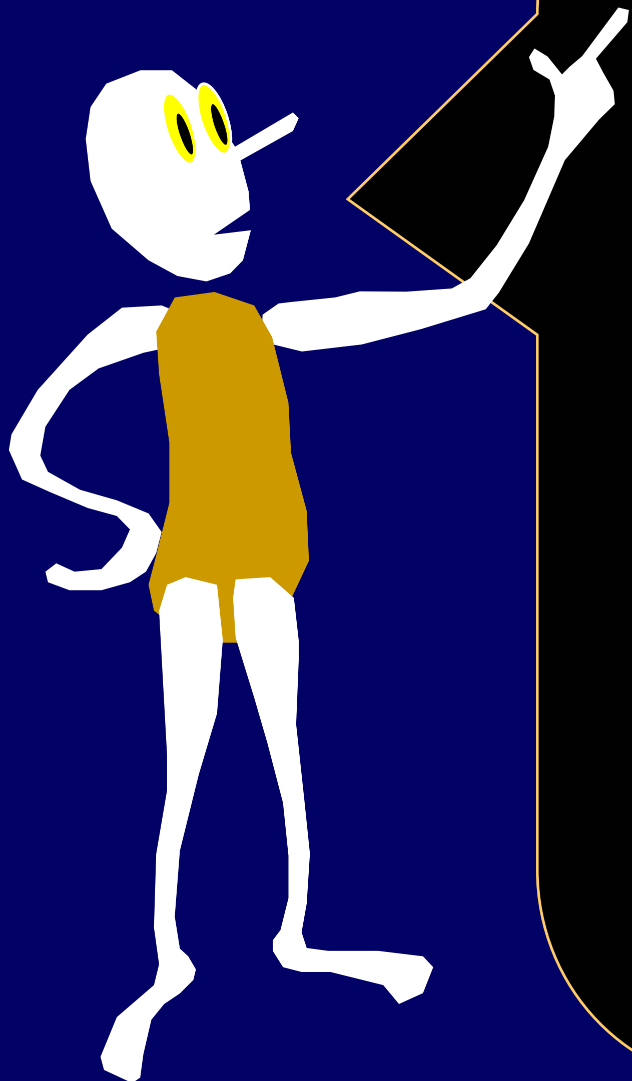
Let n be the smallest unary number such that $10^n > X$, $10^{n-1} \le X$.

Let d = X DIV 10^{n-1} [Notice that $1 \le d \le 9$]

Let Y = X MOD 10^{n-1}

Define $\mathrm{Base}_{1\ \mathrm{to}\ 10}(X) \in$ NLZ to be

$\mathrm{Base}_{1\ \mathrm{to}\ 10}(d)\ \mathrm{Base}_{1\ \mathrm{to}\ 10}(Y)$



For each $n \in PU$ define its decimal representation to be $Base_{1\ to\ 10}(n)$.

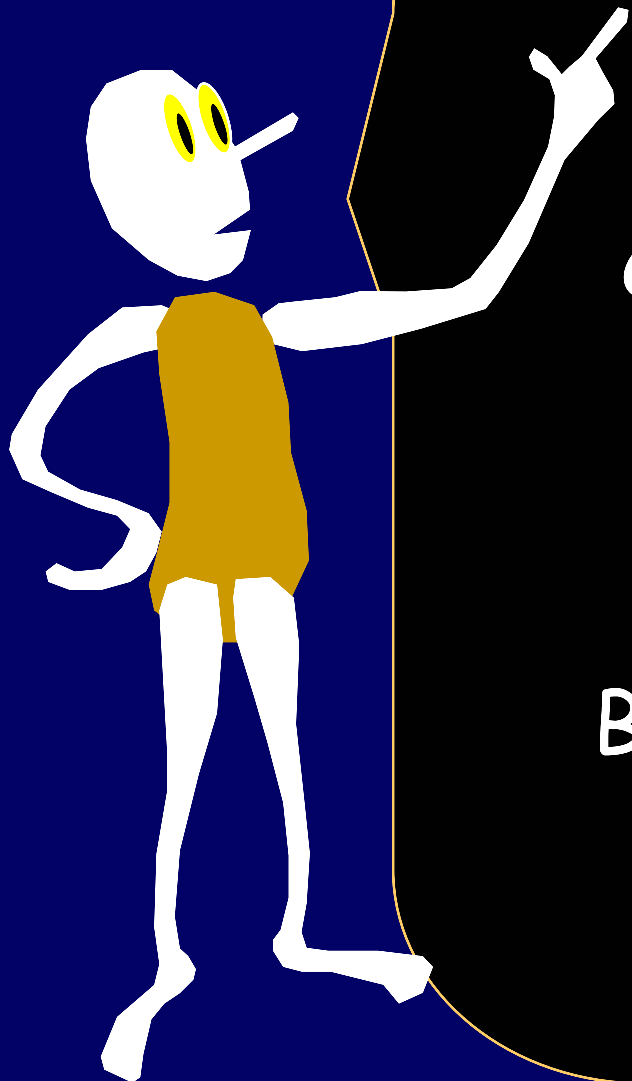
$Base_{1\ to\ 10}$ goes from PU to NLZ.

Let's verify that $\text{Base}_{1\text{ to }10}$ and $\text{Base}_{10\text{ to }1}$ really are inverse functions.

We need to show $X \in \text{NLZ} \Rightarrow \text{Base}_{1\text{ to }10}(\text{Base}_{10\text{ to }1}(X)) = X$.

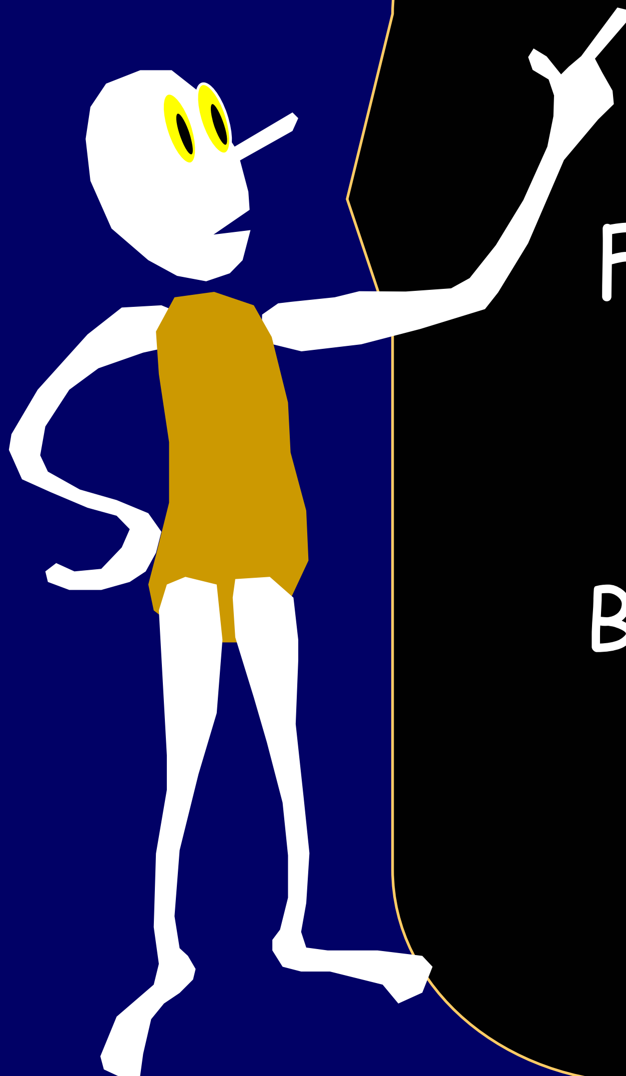
Clear when X is a single digit.

$Base_{1\text{ to }10}(Base_{10\text{ to }1}(0)) = 0$
$Base_{1\text{ to }10}(Base_{10\text{ to }1}(1)) = 1$
$Base_{1\text{ to }10}(Base_{10\text{ to }1}(2)) = 2$
$Base_{1\text{ to }10}(Base_{10\text{ to }1}(3)) = 3$
$Base_{1\text{ to }10}(Base_{10\text{ to }1}(4)) = 4$
$Base_{1\text{ to }10}(Base_{10\text{ to }1}(5)) = 5$
$Base_{1\text{ to }10}(Base_{10\text{ to }1}(6)) = 6$
$Base_{1\text{ to }10}(Base_{10\text{ to }1}(7)) = 7$
$Base_{1\text{ to }10}(Base_{10\text{ to }1}(8)) = 8$
$Base_{1\text{ to }10}(Base_{10\text{ to }1}(9)) = 9$



Let X be a shortest counter-example to the statement:

$$X \in NLZ \Rightarrow$$
$$Base_{1 \text{ to } 10}\,(Base_{10 \text{ to } 1}(Z)) = Z$$



For all Y shorter than X:

$$Y \in NLZ \Rightarrow$$

$$Base_{1 \text{ to } 10}\,(Base_{10 \text{ to } 1}(Y)) = Y$$

$X \in NLZ$, $n=\text{length}(Z) > 1$

Suppose $X=a_{n-1}\ a_{n-2}\ \ldots\ a_0$

$Base_{10\ to\ 1}(X) = Y =$

$\quad \Sigma_{i<n}\ Base_{10\ to\ 1}(a_i)*10^i$

$Base_{1\ to\ 10}\ (Y) = ?$

n is smallest PU s.t. $10^{n-1} \leq Y < 10^n$

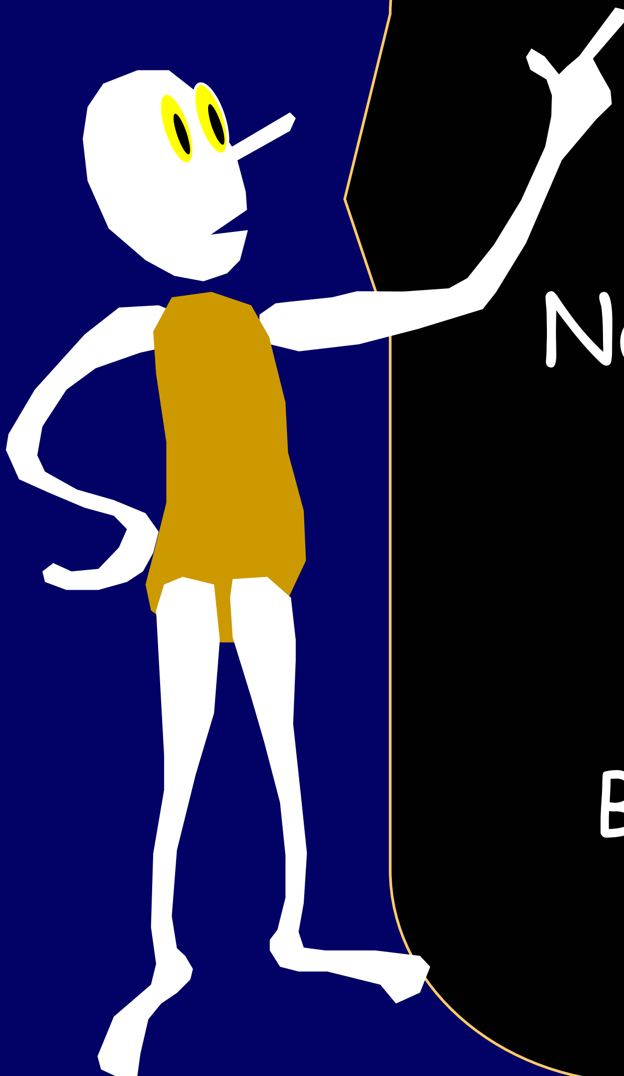
Calculate $d = Y$ DIV 10^{n-1}, $Z= Y$ MOD 10^{n-1}

$d= a_{n-1}$, $Z= \Sigma_{i<n-1}\ Base_{10\ to\ 1}(a_i)*10^i$

Output $a_{n-1}\ Base_{1\ to\ 10}\ (Z)$ [Z shorter than X]

$\quad = a_{n-1}\ a_{n-2}\ \ldots\ a_1\ a_0$

Contradiction of X being a counter-example.



No counter-example means that

For all $Y \in NLZ$,

$$\mathrm{Base}_{1 \text{ to } 10}(\mathrm{Base}_{10 \text{ to } 1}(Y)) = Y$$

So we can move back and forth between representing a number in PU or in NLZ.

Each string in PU corresponds to one and only one string in NLZ.

Base X Notation

Let Σ be an alphabet of size X. A base X digit is any element of Σ.

Let $S = a_{n-1}, a_{n-2}, \ldots, a_0$ be a sequence of base X digits.

Let $Base_{X \text{ to } 1}(S) = a_{n-1} X^{n-1} + \ldots a_2X^2 + a_1X + a_0$

S is called the base X representation of the number $Base_{X \text{ to } 1}(S)$.

$$S = a_{n-1}, a_{n-2}, \ldots, a_0$$

represents the number:

$$a_{n-1} X^{n-1} + a_{n-2} X^{n-2} + \ldots + a_0 X^0$$

Base 2 [Binary Notation]

101 represents $1\,(2)^2 + 0\,(2^1) + 1\,(2^0)$

Base 7

015 represents $0\,(7)^2 + 1\,(7^1) + 5\,(7^0)$

$$S = a_{n-1}, a_{n-2}, \ldots, a_0$$

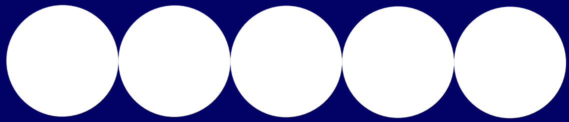
represents the number:

$$a_{n-1} X^{n-1} + a_{n-2} X^{n-2} + \ldots + a_0 X^0$$

Base 2 [Binary Notation]

101 represents $1\,(2)^2 + 0\,(2^1) + 1\,(2^0)$

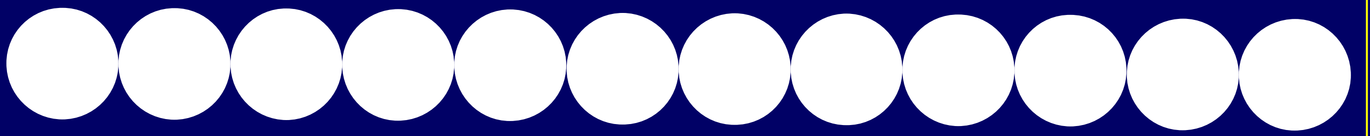
=



Base 7

015 represents $0\,(7)^2 + 1\,(7^1) + 5\,(7^0)$

=



Bases In Different Cultures

Sumerian-Babylonian: 10, 60, 360
Egyptians: 3, 7, 10, 60
Africans: 5, 10
French: 10, 20
English: 10, 12,20

Biggest n "digit" number in base X would be:

$$(X-1)X^{n-1} + (X-1)X^{n-2} + \ldots + (X-1)X^{0}$$

Base 2

111 represents $1\,(2)^2 + 1\,(2^1) + 1\,(2^0)$

Base 7

666 represents $6\,(7)^2 + 6\,(7^1) + 6\,(7^0)$

Biggest n "digit" number in base X would be:

$$(X-1)X^{n-1} + (X-1)X^{n-2} + \ldots + (X-1)X^{0}$$

Base 2

111 represents $1\,(2)^2 + 1\,(2^1) + 1\,(2^0)$

$111 + 1 = 1000$ represents 2^3

Thus, 111 represents $2^3 - 1$

Base 7

666 represents $6\,(7)^2 + 6\,(7^1) + 6\,(7^0)$

$666 + 1 = 1000$ represents 7^3

Thus, 666 represents $7^3 - 1$

Biggest n "digit" number in base X would be:

$$S= (X-1)X^{n-1} + (X-1)X^{n-2} + \ldots + (X-1)X^0$$

\Add 1 to get: $X^n + 0\, X^{n-1} + \ldots + 0\, X^0$

Thus, $S = X^n - 1$

Base 2

111 represents $1\,(2)^2 + 1\,(2^1) + 1\,(2^0)$

$111 + 1 = 1000$ represents 2^3

Thus, 111 represents $2^3 - 1$

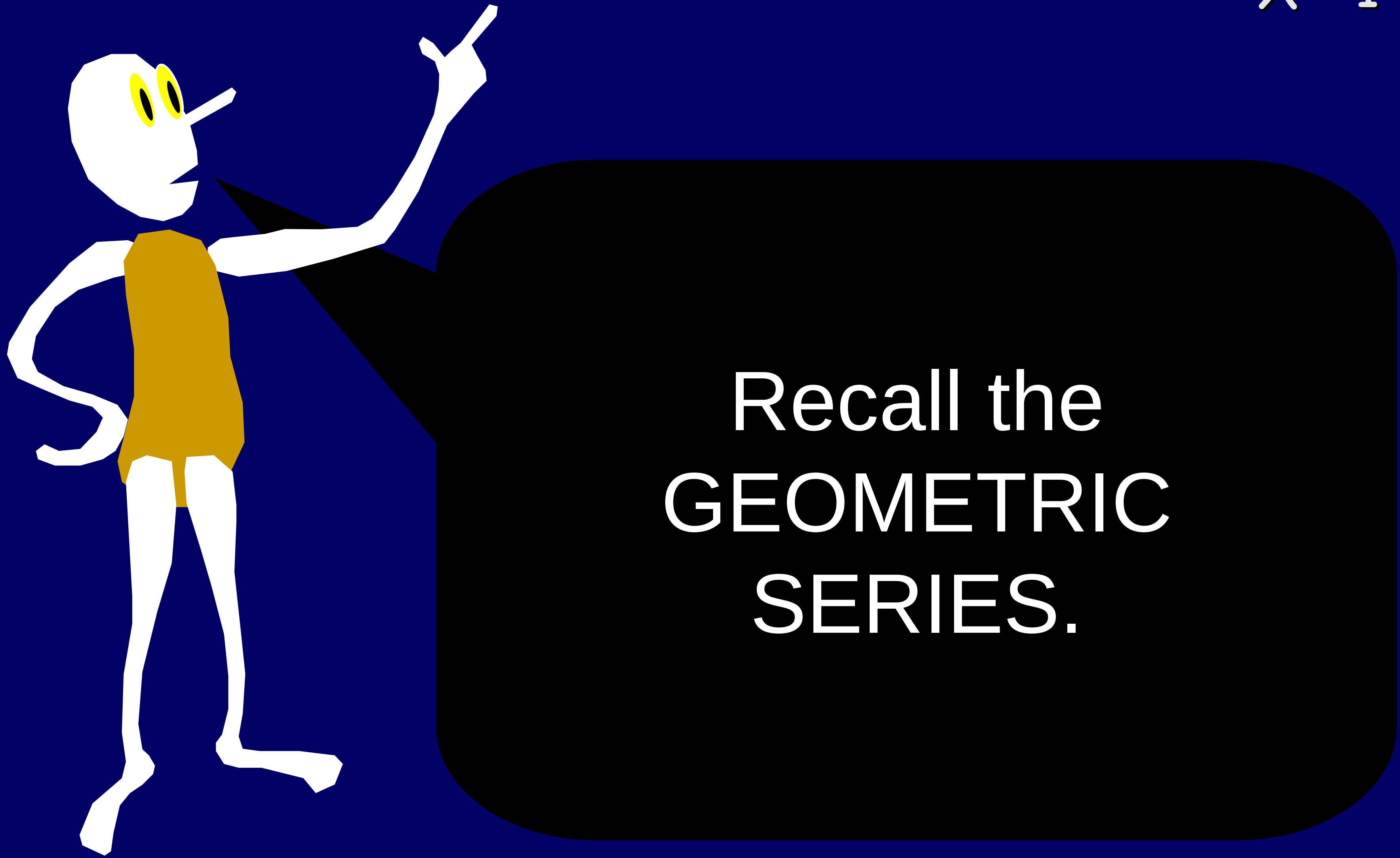
Base 7

666 represents $6\,(7)^2 + 6\,(7^1) + 6\,(7^0)$

$666 + 1 = 1000$ represents 7^3

Thus, 666 represents $7^3 - 1$

$$1 + X^1 + X^2 + X^3 + \dots + X^{n-2} + X^{n-1} = \frac{X^n - 1}{X - 1}$$



$$(X-1)X^{n-1} + (X-1)X^{n-2} + \ldots + (X-1)X^{0}$$

$$= X^{n} - 1$$

Proof:

Factoring out (X-1), we obtain:

$(X-1)\,[X^{n-1} + X^{n-2} + \ldots\ldots + X + 1] =$

$(X-1)\,[(X^{n} - 1)/(X-1)] =$

$X^{n} - 1$

The highest n digit number in base X.

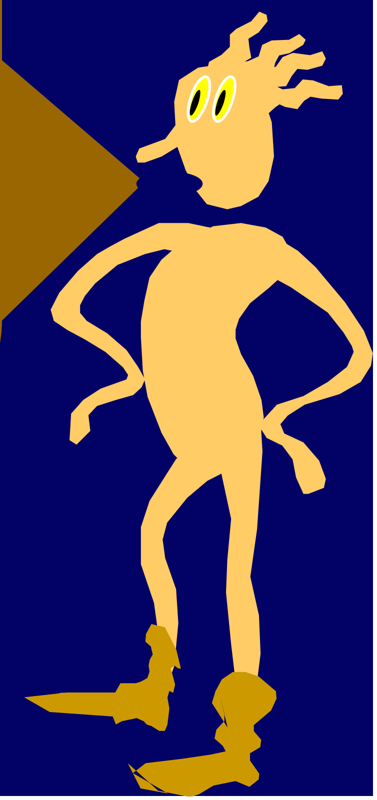
$$(X-1)\,(1 + X^1 + X^2 + X^3 + \dots + X^{n-1}) = X^n - 1$$

Base X. Let S be the sequence of n digits each of which is X-1. S is the largest number of length n that can be represented in base X.

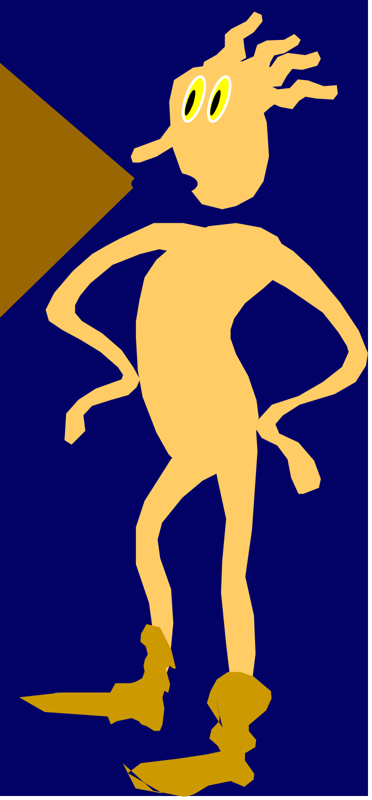
$$S = X^n - 1$$

Each of the numbers from 0 to X^n-1 is uniquely represented by an n-digit number in base X.

We could prove this using our previous method, but let’s do it another way.



Our proof introduce an unfamiliar kind of base representation.



Plus/Minus Base X

An plus/minus base X digit is any integer $-X < a < X$

Let $S = a_{n-1}, a_{n-2}, \ldots, a_0$ be a sequence of plus/minus base X digits. S is said to be the plus/minus base X representation of the number:

$$a_{n-1} X^{n-1} + a_{n-2} X^{n-2} + \ldots + a_0 X^0$$

Does each n-digit number in plus/minus base X represent a different integer?

No.
Consider plus/minus binary.

5 = 0 1 0 1 = 1 0 -1 -1

Humm..
So what is it good for?

Not every number has a unique plus/minus binary representation, but 0 does.

0 has a unique n-digit plus/minus base X representation as all 0's.

Suppose $a_{n-1} X^{n-1} + a_{n-2} X^{n-2} + \ldots + a_0 X^0 = S = 0$, where there is some highest k such that $a_k \neq 0$. Wl.o.g. assume $a_k > 0$.

Because $X^k > (X-1)(1 + X^1 + X^2 + X^3 + \ldots + X^{k-1})$
no sequence of digits from a_{k-1} to a_0 can represent a number as small as $-X^k$

Hence $S \neq 0$. Contradiction.

Each of the numbers from 0 to X^n-1 is uniquely represented by an n-digit number in base X.

We already know that n-digits will represent something between 0 and $X^n - 1$.

Suppose two distinct sequences represent the same number:

$$a_{n-1} X^{n-1} + a_{n-2} X^{n-2} + \ldots + a_0 X^0 =$$
$$b_{n-1} X^{n-1} + b_{n-2} X^{n-2} + \ldots + b_0 X^0$$

The difference of the two would be an plus/minus base X representation of 0, but it would have a non-zero digit. Contradiction.

Each of the numbers from 0 to X^n-1 is uniquely represented by an n-digit number in base X.

n digits represent up to $X^n - 1$
n-1 digits represents up to $X^{n-1} - 1$

Let k be a number: $X^{n-1} \leq k \leq X^n - 1$
So k can be represented with n digits.

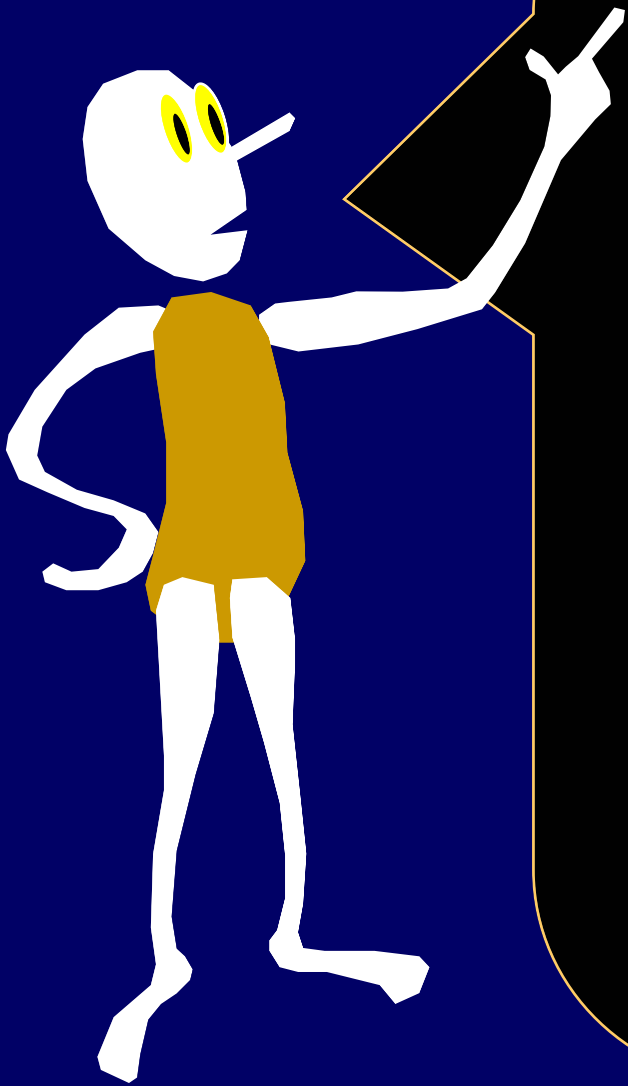
For all k: $\lfloor \log_x k \rfloor = n - 1$

So k uses $\lfloor \log_x k \rfloor + 1$ digits.

Fundamental Theorem For Base X:

Each of the numbers from 0 to X^n-1 is uniquely represented by an n-digit number in base X.

$\boxed{k \text{ uses } \lfloor \log_x k \rfloor + 1 \text{ digits in base } X.}$



n has length n in unary, but has length $\lfloor \log_2 n \rfloor + 1$ in binary.

Unary is exponentially longer than binary.

Egyptian Multiplication

The Egyptians used decimal numbers but multiplied and divided in binary

Egyptian Multiplication a times b by repeated doubling

b has some n-bit representation: $b_n..b_0$

Starting with a,
repeatedly double largest so far to obtain: $a, 2a, 4a, ..., 2^n a$

Sum together all $2^k a$ where $b_k = 1$

Egyptian Multiplication 15 times 5 by repeated doubling

5 has some 3-bit representation: 101

Starting with 15,
repeatedly double largest so far to obtain: 15, 30, 60

Sum together all $2^k(15)$ where $b_k = 1$:

$$15 + 60 = 75$$

Why does that work?

$$b = b_0 2^0 + b_1 2^1 + b_2 2^2 + \ldots + b_n 2^n$$
$$ab = b_0 2^0 a + b_1 2^1 a + b_2 2^2 a + \ldots + b_n 2^n a$$

If b_k is 1 then $2^k a$ is in the sum.
Otherwise that term will be 0.

Wait! How did the Egyptians do the part where they converted b to binary?

They used repeated halving to do base conversion. Consider …

Egyptian Base Conversion

Output stream will print right to left.
Input X.
Repeat until X=0
{
    If X is even then Output 0;
        Otherwise {X:=X-1; Output 1}

    X:=X/2
}

Egyptian Base Conversion

Output stream will print right to left.
Input X.
Repeat until X=0
{
    If X is even then Output 0;
        Otherwise Output 1

    X:= $\lfloor X/2 \rfloor$
}

Start the algorithm

010101

1

Repeat until X=0

{ If X is even then Output 0;

Otherwise Output 1;

$X := \lfloor X/2 \rfloor$

}

Start the algorithm

01010

1

Repeat until X=0

{ If X is even then Output 0;

Otherwise Output 1;

$X := \lfloor X/2 \rfloor$

}

Start the algorithm

01010

01

Repeat until X=0

{ If X is even then Output 0;
Otherwise Output 1;
$X := \lfloor X/2 \rfloor$

}

Start the algorithm

0101

01

Repeat until X=0

{ If X is even then Output 0;

Otherwise Output 1;

$X := \lfloor X/2 \rfloor$

}

Start the algorithm

0101

101

Repeat until X=0

{ If X is even then Output 0;
Otherwise Output 1;
$X := \lfloor X/2 \rfloor$

}

Start the algorithm

010

101

Repeat until X=0

{ If X is even then Output 0;

Otherwise Output 1;

$X := \lfloor X/2 \rfloor$

}

And Keep Going until 0

0

010101

Repeat until X=0

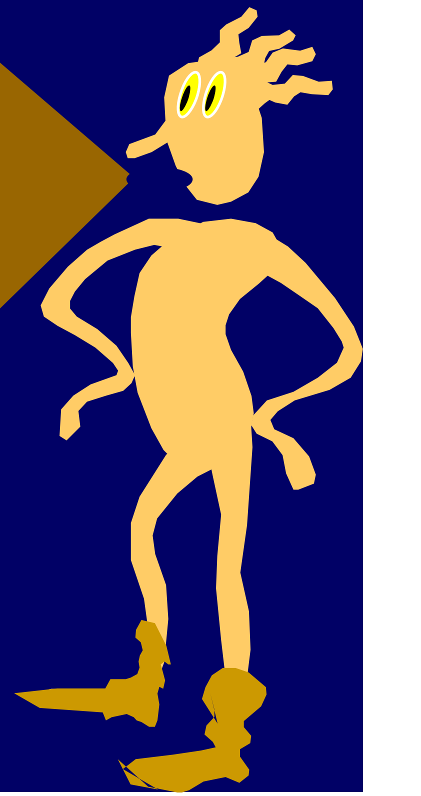
{ If X is even then Output 0;

Otherwise Output 1;

$X := \lfloor X/2 \rfloor$

}

Sometimes the Egyptian combined the base conversion by halving and the multiplication by doubling into one algorithm



Rhind Papyrus (1650 BC)
70*13

70	13	*	70
140	6		
280	3	*	350
560	1	*	910

Rhind Papyrus (1650 BC)
70*13

70	13 *	70
140	6	
280	3 *	350
560	1 *	910

Binary for 13 is 1101 = $2^3 + 2^2 + 2^0$

$70*13 = 70*2^3 + 70*2^2 + 70*2^0$

Rhind Papyrus (1650 BC)

17	1
34	2 *
68	4
136	8 *

184 48 14

Rhind Papyrus (1650 BC)

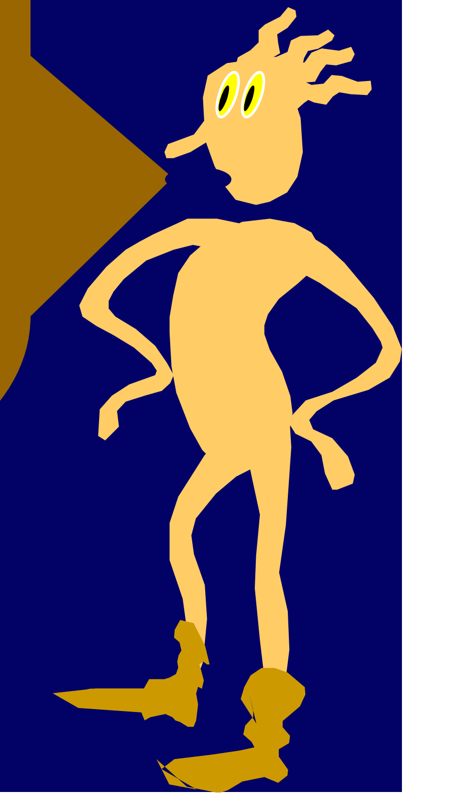
17	1
34	2 *
68	4
136	8 *

184 48 14

184 = 17*8 + 17*2 + 14

184/17 = 10 with remainder 14

This method is called “Egyptian Multiplication/Division” or “Russian Peasant Multiplication/Division”.




Wow. Those Russian peasants were pretty smart.

Standard Binary Multiplication
= Egyptian Multiplication

```
          * * * * * * * *
X                     101
-------------------------
           * * * * * * * *
       * * * * * * * *
-------------------------
     * * * * * * * * * * *
```

Egyptian Base 3

We have defined

Base 3: Each digit can be 0, 1, or 2

Plus/Minus Base 3 uses -2, -1, 0, 1, 2

Here is a new one:

Egyptian Base 3 uses -1, 0, 1

Example: 1 -1 -1 = 9 - 3 - 1 = 5

Unique Representation Theorem for Egyptian Base 3

No integer has 2 distinct, n-digit, Egyptian base-3 representations. We can represent all integers from
$-(3^n-1)/2$ to $(3^n-1)/2$

Proof; If so, their difference would be a non-trivial plus/minus base 3 representation of 0. Contradiction.

Highest number = 1111…1 = $(3^n-1)/2$

Lowest number = -1-1-1-1…-1 = $-(3^n-1)/2$

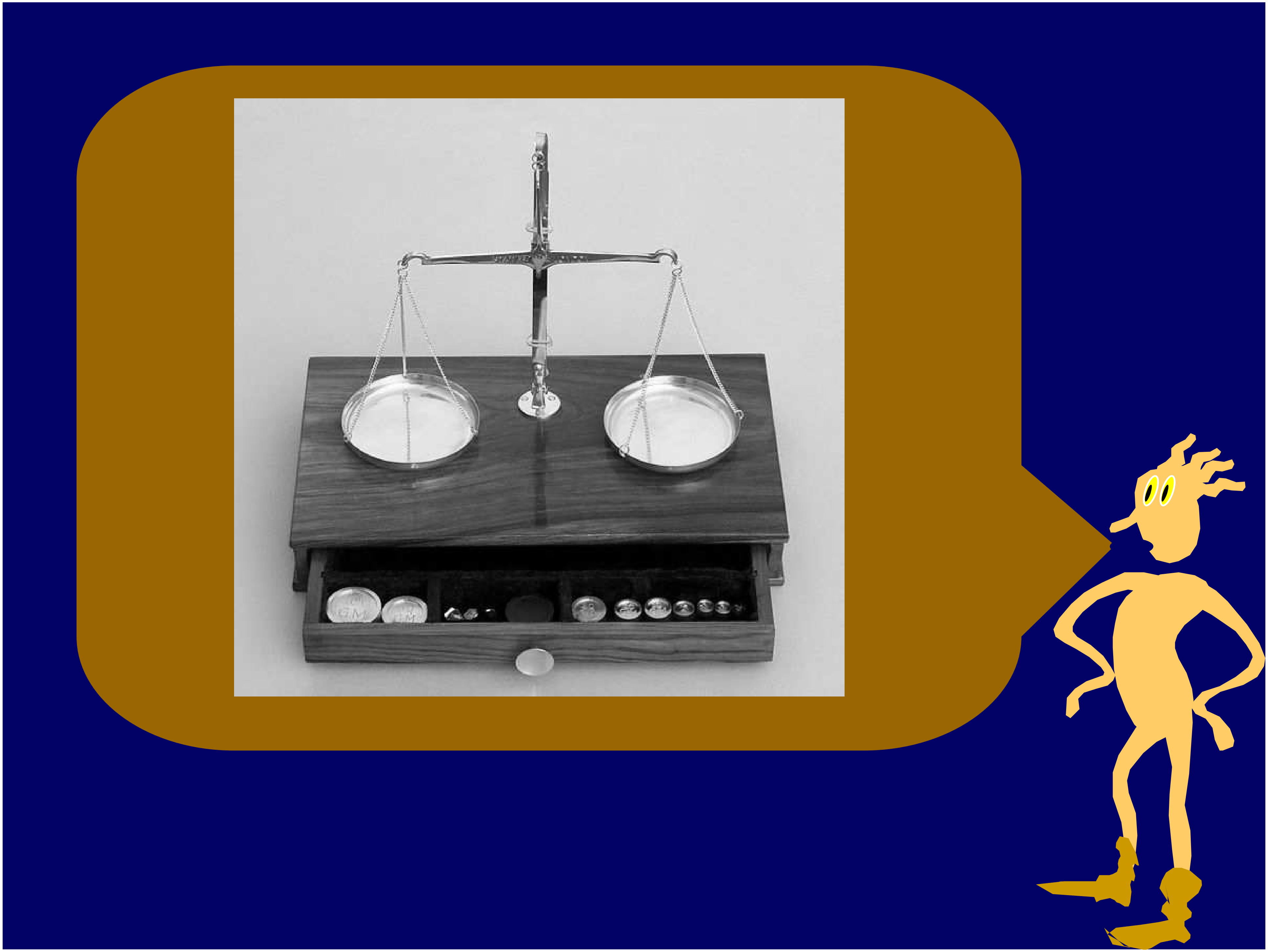
Unique Representation Theorem for Egyptian Base 3

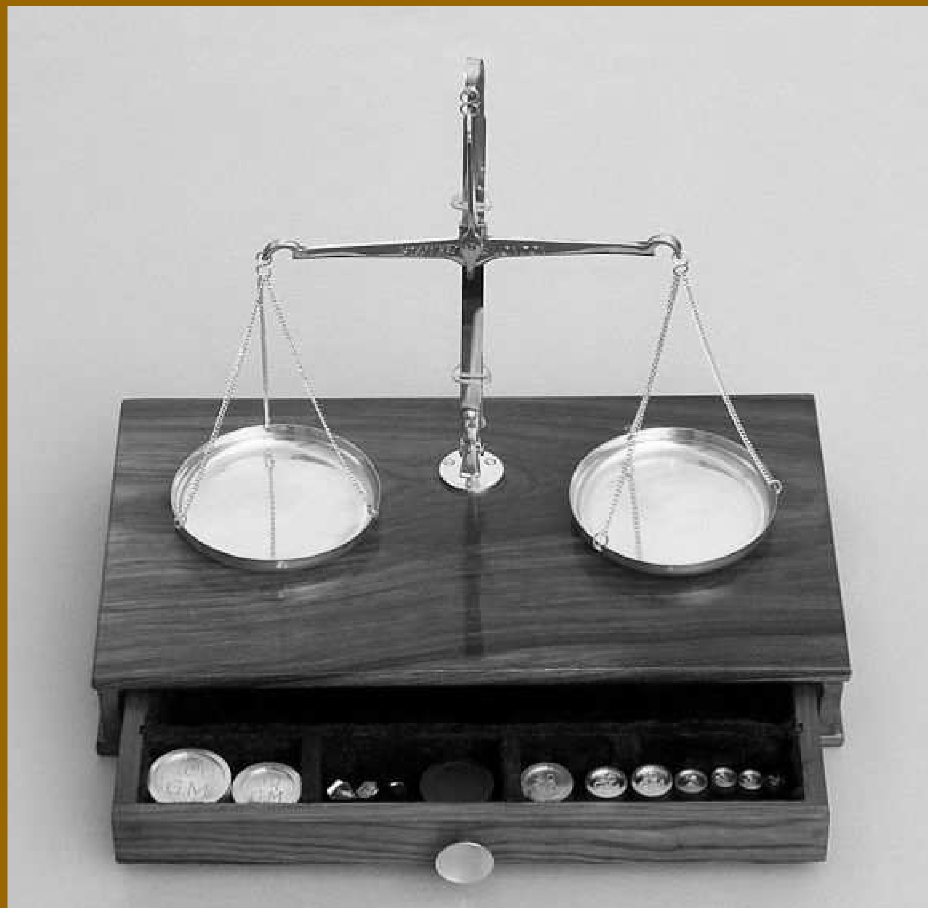
No integer has 2 distinct, n-digit, Egyptian base-3 representations. We can represent all integers from $-(3^n-1)/2$ to $(3^n-1)/2$

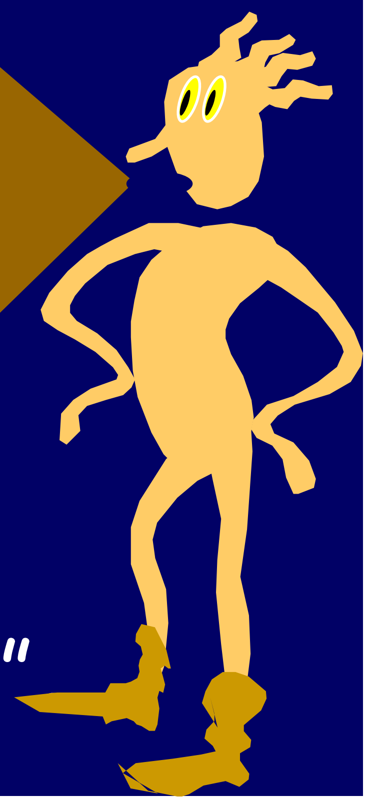
3^n, n-digit, base 3 representations of the numbers from 0 to $3^n - 1$

Subtract 111111…111 = $(3^n - 1)/2$ from each to get an Egyptian base 3 representation of all the numbers from $-(3^n-1)/2$ to $(3^n-1)/2$.

How could this be Egyptian? Historically, negative numbers first appear in the writings of the Hindu mathematician Brahmagupta (628 AD).







One weight for each power of 3.
Left = “negative”. Right = “positive”

References

The Book Of Numbers, by J. Conway and R. Guy

History of Mathematics, Histories of Problems, by The Inter-IREM Commission