Binomial Coefficients

Definition: The binomial coefficient $\binom{n}{k} = \dfrac{n!}{k!(n-k)!}$ for all $n \geq k \geq 0$ (read "n choose k")
where $n! = n \cdot (n-1) \cdot (n-2) \cdot \ldots \cdot 2 \cdot 1$ and, by definition, $0! = 1$.
This counts the number of k-element subsets of an n-element set, or equivalently, the number of ways to choose a set of k elements, taken from a set of n elements.

Example 1: Calculate: $\binom{10}{3} = \dfrac{10!}{3!\ 7!} = \dfrac{10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1}{3 \cdot 2 \cdot 1 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1} =$

Example 2: How many 8-bit strings contain exactly 3 zeros?

Example 3: Use the definition to simplify each of the following:

$\binom{n}{0} =$ $\qquad\qquad$ $\binom{n}{n} =$

Example 4: Suppose a string of n characters contains only two possible characters: * and |
Write an expression for the number of n-bit strings that contain exactly 2 *'s. Simplify your expression to be written as a polynomial.

Pascal's Identity: For all $n \geq k \geq 1$, $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

Pascal's Triangle

(see https://brilliant.org/wiki/pascals-triangle/)

Logarithms

logs and exponents are inverse operations

- To "undo" multiplication, you divide.
- To "undo" addition, you subtract.
- To "undo" exponentiation, you hit it with a log.

Definition $y = \log_b x \Leftrightarrow b^y = x$ Notation: We use lg x to represent $\log_2 x$.

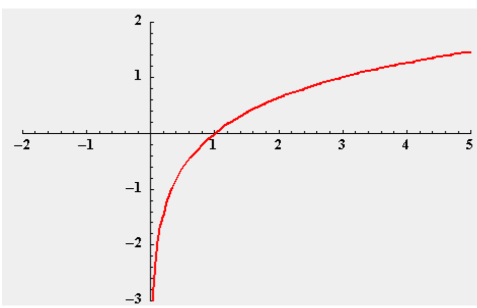
Powers of 2: $2^0 = 1$, $2^1 = 2$, $2^2 = 4$,

Example 1: Use the definition to determine each of the following:

lg 32 = lg 1024 = lg 1 =

$\log_3 81$ = log 100,000 =

Graph of $y = \log_b x$



Note: $\log_b 1 = 0$ for any base $b > 0$ (since $b^0 = 1$).
The logarithm function has domain $\{x \mid x > 0\}$. The x-intercept is at $x = 1$.
The logarithm function is a strictly increasing and very slowly growing function.

Example 2: Use powers of 2 to calculate:

$\lfloor \lg 70 \rfloor =$ $\lceil \lg 1100 \rceil =$

Change of Base Formula: $$\log_a x = \frac{\log_b x}{\log_b a}$$

Example 3: Use a calculator and the change of base formula to calculate each of the following:

$\log_3 687 =$

$\log_5 791 =$

Properties of Logs:

$$log_a(xy) = log_a x + log_a y$$

$$log_a\left(x^r\right) = r log_a x$$

Example 4: Use properties of logs to simplify each of the following:

$\lg(8n)$

$\lg(n^2)$

$\lg(n^{10})$

$\lg(\sqrt{n})$

$\lg(n^n)$

Example 5: The algorithm binarySearch is used to find the index for a `key` value in a sorted array `a` of `n` items. Determine the number of times the while loop is executed in the worst case.

```
binarySearch(a, key)
{
  left = 1
  right = a.last
  while (left <= right)
  {
     mid = (left + right) / 2
     if (key < a[mid])
        right = mid - 1
     else if (key > a[mid])
        left = mid + 1
     else
        return mid  // found
  }
  return -1         // not found
}
```

search 3770 student records => $\lg(3770) \approx 11.8$

search 332,000,000 people in US => $\lg(332{,}000{,}000) \approx 28.3$