Announcements

• Homework 5 is due now.

• Homework 6 is due next Monday, April 6 at 10am in class.
  • It will be posted on the web site today.

• We will not cover structural induction on strings.
  • It will not be on the exam or homework.
Last Time – Full Binary Trees

• Recursive definition:
  • **Basis step:** A single vertex r is a full binary tree
  • **Recursive step:** If $T_1$ and $T_2$ are full disjoint full binary trees, there is a full binary tree, denoted $T_1 \bullet T_2$, consisting of a root r with edges connecting to the roots of $T_1$ and $T_2$. 
Use structural induction to show that $N(T) \geq 2H(T) + 1$, where $T$ is a full binary tree.
Use structural induction to show that \( N(T) \geq 2H(T) + 1 \), where \( T \) is a full binary tree.
Use structural induction to show that \( N(T) \geq 2H(T) + 1 \), where \( T \) is a full binary tree.
RECURSIVE ALGORITHMS

Section 5.4
Example of a Recursive Algorithm

```plaintext
procedure mystery(n: nonnegative integer)
    if n = 0 then return 1
    else return n*mystery(n-1)
```
Definition of a Recursive Algorithm

• An algorithm is called **recursive** if it solves a problem by reducing it to an instance of the same problem with smaller input.

• For the algorithm to terminate, this instance must be reduced to some initial case(s) for which the solution is known.

```plaintext
procedure mystery(n: nonnegative integer)
    if n = 0 then return 1
    else return n*mystery(n-1)
```
Another Example

• Find a recursive algorithm for computing $a^n$, where $a$ is a non-zero real number and $n$ is a nonnegative integer.
• First, what is a recursive definition of $a^n$?

• \textit{procedure} power(a: nonzero real number, 
n: nonnegative integer)
procedure \textit{power}(a: nonzero real number, \
    n: nonnegative integer) \
  
  if \( n = 0 \) then return 1 \
  else return \( a \times \text{power}(a, n-1) \)

Prove that the algorithm is correct, i.e., for inputs \( a \) and \( n \), the output is \( a^n \).
Fibonacci – Recursive vs. Iterative

• Recall the recursive definition of the sequence of Fibonacci numbers:
  • Initial Conditions: \( f_0 = 0, f_1 = 1 \)
  • Recurrence Relation: \( f_n = f_{n-1} + f_{n-2} \)
• Write a recursive algorithm that outputs the \( n^{th} \) Fibonacci number.

\[
\text{procedure } \text{fibonacci}(n: \text{ nonnegative integer})
\]
Recursive Fibonacci

```
procedure fibonacci(n: nonnegative integer)
if n = 0 then return 0
else if n=1 then return 1
else return fibonacci(n-1) + fibonacci(n-2)
```

- How many additions does it take to compute the $n^{th}$ Fibonacci number $f_n$?
Iterative Fibonacci Algorithm

**Procedure** iterative_fib(n: nonnegative integer)

* if n = 0 then return 0
  * else
    * x := 0
    * y := 1
    * for i=1 to n-1
      * z := x + y
        * x := y
        * y := z
    * return y

• How many additions does it take to compute the n\textsuperscript{th} Fibonacci number?
Counting Things

• How many bit strings of length 7 are there?

• There are seven “positions”.
• Each position has two possible assignments, 0 or 1.
• The total number of bit strings $= 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 = 2^7$
Basic Counting Principles: The Product Rule

• The Product Rule:

  Suppose a procedure can be broken down into a sequence of two tasks. There are $n_1$ ways to do the first task and $n_2$ ways to do the second task. Then there are $n_1 \cdot n_2$ ways to do the procedure.

• For a bit string of length 7, there are 7 tasks. Each task has 2 possible values.
Example of the Product Rule

• How many different license plates can be made if each plate contains a sequence of three uppercase English letters followed by three digits?
The Product Rule in Terms of Sets

- If we have $m$ tasks, and the possibilities for task $i$ are contained in the set $A_i$, how many total possibilities are there?

$$|A_1 \times A_2 \times \cdots \times A_m| = |A_1| \cdot |A_2| \cdot \cdots \cdot |A_m|.$$
Counting Functions

• How many functions are there from a set with $m$ elements to a set with $n$ elements?

• **Solution:** To define a function, for each of the $m$ elements in the domain, we choose one of $n$ elements in the codomain to map to. The product rule tells us that there are $n \cdot n \cdot \cdots \cdot n = n^m$ such functions.

• Assuming $n \geq m$, how many injective functions are there?
  • **Solution:** $n \cdot (n-1) \cdot (n-2) \cdot \cdots \cdot (n - m + 1)$
Counting Subsets of a Finite Set

• Use the product rule to prove that a set with n elements has $2^n$ subsets.

• List the elements of the set in some arbitrary order.

• Make a bit string of length n. Each bit corresponds to a single element of the set.

• How can we represent subsets using this bit string?
  • 1 means element is in the set; 0 means element is not in the set.

• How many possible bit strings are there?
  • This equals the number of subsets.
Basic Counting Principles: The Sum Rule

• A student can choose a project from one of three lists. The first list contains 23 projects.
• the second contains 15 projects,
• and the third contains 20 projects.
  How many projects are there to choose from?

• The Sum Rule: If a task can be done either in one of \( n_1 \) ways or in one of \( n_2 \) ways, where no one of the \( n_1 \) ways is the same as any of the \( n_2 \) ways, then there are \( n_1 + n_2 \) ways to do the task.
Combining the Sum and Product Rule

• Suppose a programming language has the following naming convention for variables:
  • A variable name can be either a single lowercase letter or a lowercase letter followed by a single digit.

• What is the total number of possible variable names?
Counting Passwords

• Each user on a computer system has a password which follows all of the following rules
  • The password is six to eight characters long
  • Each character is an uppercase letter or a digit.
  • Each password contains at least one digit.

• How many possible passwords are there?
Basic Counting Principles: Subtraction Rule

• **Subtraction Rule**: If a task can be done either in one of \( n_1 \) ways or in one of \( n_2 \) ways, then the total number of ways to do the task is \( n_1 + n_2 \) minus the number of ways to do the task that are common to the two different ways.

• Also known as, the principle of inclusion-exclusion:

\[
|A \cup B| = |A| + |B| - |A \cap B|
\]
Counting Bit Strings

How many bit strings of length eight start with a 1 bit or end with the two bits 00?
Basic Counting Principles: Division Rule

- **Division Rule**: Suppose a task can be carried out in \( n \) ways, but for any way \( w \), there are \((d-1)\) ways that are considered identical to \( w \). Then, there are \( n/d \) ways to do the task.

- How many ways are there to seat four people around a circular table, where two seatings are considered the same when each person has the same left and right neighbor?
Good Problems to Review

- Section 5.4: 7, 11, 13, 21, 32
- Section 6.1: 3, 5, 11, 17, 33, 35, 45