CS 281
Prof. Alex Pothen
Webbook:

www.cs.odu.edu/~webbook
www.cs.odu.edu/~pothen/Courses/CS281
Mon Nov 2, 1998

Lecture 18

Sec 3.4 Recursive Algorithms
Sec 2.4 Euclidean Algorithm

Next Lecture: Sec 6.1 Relations
(Background: Sec 1.4, 1.5 Sets)
Recursive definitions

- A recursive definition involves
  (1) one or more basis rules, which define some simple objects, and
  (2) one or more inductive rules, whereby larger objects are defined in terms of smaller objects in the collection.

- Let a set $S$ on the integers be defined by:
  1. $1 \in S$;
  2. if $x, y \in S$, then $x + 2$ and $x - 2$ also belong to $S$.

  What numbers does the set $S$ consist of?

- Give a recursive definition for the set of even, nonnegative integers.
Character Strings
Recursive definition

- Consider strings over the alphabet of characters

\[ a < b < \ldots < z. \]

- Recursive definition:
  1. The empty string \( \lambda \) is a string.
  2. If \( w \) is a string, then \( wx \) is a string, where \( x \) is any character in the alphabet.

- We can build up any character string from these rules.
  Suppose in (2) we restrict \( x \) to the letter \( a \). What strings can we build with this rule?
Character Strings II
Recursive definition

- Consider strings over the alphabet of characters
  
  \[ 0 < 1. \]

- Recursive definition:
  (1) The empty string \( \lambda \) is a string.
  (2) If \( w \) is a string, then \( 0w1 \) is a string.

Describe this set of strings.
Lexicographic Orderings

- Lexicographic (dictionary) ordering of strings

- Basis:
  \( \lambda < w \) for any nonempty string \( w \).
  If \( c < d \), where \( c \), \( d \) are characters, then for any strings \( w \), \( x \), we have \( cw < dx \).

- Inductive Step:
  If \( w < x \) for two strings \( w \), \( x \), then \( cw < cx \) for any character \( c \).

- The induction is over the number of characters that are equal at the beginning of the two strings.
Lexicographic Orderings II

- Example: ball < bat.

  ll < t
  all < at
  ball < bat

- Example: bat < batter.

  λ < ter
  t < tter
  at < atter
  bat < batter
Divisors and Modulo Arithmetic

• Let $a$ be an integer, and $d$ a positive integer. Then if we divide $a$ by $d$, we compute a quotient $q$ and a remainder $r$ such that $a =qd + r$. The remainder $r$ is nonnegative, and less than $d$.

• Examples: Let $a = 14$, $d = 4$.

  $14 = 3 \times 4 + 2$
  $-11 = (-3) \times 4 + 1$.

• Sometimes we wish to know only the remainder: Then $a \mod d = r$ (modulo arithmetic).

• If $r = 0$, then $d$ is a divisor of $a$. 
Greatest Common Divisor

- The largest integer \(d\) that is a divisor of \(a\) is the greatest divisor of \(a\).

- Let \(a\) and \(b\) be both non-negative integers, and let at least one of them be not zero. The largest integer that is a divisor of \(a\) and \(b\) is called the greatest common divisor of \(a\) and \(b\): \(\text{gcd}(a, b)\).

- What is \(\text{gcd}(75, 45)\)?

- What is \(\text{gcd}(45, 0)\)?
Greatest Common Divisor
Euclidean Algorithm

GCD := proc(a::integer, b::integer)
x := a; y := b;
while (y $\neq$ 0) do
    rem := x mod y;
    x := y; y := rem;
od;
Return (x);
Euclidean Algorithm
Recursive Algorithm

GCD := proc(a::integer, b::integer)
{Assume that $a > b$
if $b = 0$ then gcd$(a, b) := a$;
else gcd$(a, b) :=$ gcd$(b, a \mod b)$;
fi