Discrete Math

Lecture 35

Algorithms &
Pseudocode

Def: an algorithm is a finite seq. of instructions
that, when followed, may help perform a task or
solve a problem

- often written in a computer language

pseudo-code is a description of an algorithm
designed to be read by humans

- often uses common English
- also uses slightly technical terms

Common
English

C, C++

ex) task: find the smallest element in a (finite) $S \subseteq \mathbb{N}$

$$S = \{n_1, n_2, n_3, \ldots, n_k\}$$

set $MIN = n_1$

for $i = 2$ to $i = k$

if $MIN > n_i$, then $MIN \to n_i$

return MIN

Algorithms

- take in inputs
- produce outputs ← correct solutions to problems
- well-defined steps
- finite # of steps   (could be VERY big!!)

we'd <u>love</u> to minimize this

and will focus on estimating these soon!

ex]   linear search   X

list of integers $a_1, a_2, \ldots, a_n$ $(a_i \in \mathbb{Z})$

given x

while $(i \leq n$ and $x \neq a_i)$

$\qquad i \to i+1$

$\qquad$ if $i = n$, return "x is not in list"

$\qquad$ else   return x

ex]   an algorithm for computing remainders

(note: most languages use "%" to do this)

(ex: $5 \% 4$ returns 1 :   5 mod 4 $= 1$)

$\qquad$ given $a > 0$, $n > 0$

$\qquad$ while $a - q \cdot n \geq 0$

$\qquad\qquad q \to q+1$

$\qquad$ return $a - (q-1) \cdot n$

lets call this "%"

Euclidean Algorithm

$a > 0, \quad b > 0$

$$a = b \cdot q + r$$

$$b = \_ r + \_$$

while $b \neq 0,$

$r \rightarrow a \% b$

$a \rightarrow b$

$b \rightarrow r$

return b