Announcements

Midterm 1 Wed in-class (50 minutes)

Reference sheet allowed (one A4 sheet, both sides)

Sections covered: 2.1-3, 3.1-2, 5.1-2, 6.1-2

Problem session → Review session on Tues. (time/location TBD)

Practice problems partial sol'ns posted

See policy email for more

Midterm 1 review:

(Partial) list of topics:

Sets

- Roster notation vs. set builder notation
- Special sets ($\mathbb{Z}$, $\mathbb{R}$, $\emptyset$, etc.)
- Venn diagrams
- Subset, power set, Cartesian product
- Cardinality
- Set operations: union, intersection, set-minus, complement
- Set identities (1–10)
- Proof techniques: elt. chasing, membership tables

Functions

- Definition
- Domain, codomain, range/image, preimage
- Injective/surjective/bijective (& pf. techniques)
- Composition
- Inverses + invertibility

Algorithms

- Definition
- Properties (describe and check)
- Perform an algorithm
- Write an algorithm
- Searching/sorting/greedy change

Big-O

- Precise def'n of O, Ω, Θ; proof techniques
- Tricks & heuristics ($1 < \log x < x < x^2 < \cdots < e^x < \cdots$)

Induction

- Mathematical vs. strong
- Base case, inductive step
- Critique proofs
- Various examples from class & H/W

Counting

Sum / product / subtraction / division rules

Combining the rules (examples from class, HW)

(Generalized) pigeonhole principle

Examples:

1) Ex 6.2.8: Telephone numbers are of the form,

$$\underbrace{NXX}_{\text{area code}} - NXX - XXXX$$

where each N can be a digit from 2 to 9 and each X can be a digit from 0 to 9.

A state has 25,000,000 phones. How many area codes does it need to ensure each phone has a diff. num?

Sol'n: $NXX - XXXX$

$8 \cdot 10 \cdot 10 \cdot 10 \cdot 10 \cdot 10 \cdot 10 = 8{,}000{,}000$ numbers per area code

$m = 25$ million, $n = 8$ million

$$\lceil m/n \rceil = \left\lceil \frac{25}{8} \right\rceil = 4$$

So we need 4 area codes.

2) Prove that $A \cup (A \cap B) = A$ for all sets A, B

Pf: We prove the result by showing that
a) $A \cup (A \cap B) \subseteq A$ and b) $A \subseteq A \cup (A \cap B)$.

a) Let $x \in A \cup (A \cap B)$. Then either $x \in A$ or $x \in A \cap B$. In the former case, clearly $x \in A$, and in the latter case, $x \in A$ and $x \in B$, so $x \in A$. Thus, $A \cup (A \cap B) \subseteq A$.

b) Let $x \in A$. Then $x \in A$, so $x \in A \cup (A \cap B)$. Thus, $A \subseteq A \cup (A \cap B)$.

$\square$

3) Consider $f: \mathbb{Z} \to \mathbb{Z}_{\geq 0}$ where $f(x) = |x| + 1$. Determine, with proof, whether f is inj., surj., bij., or none.

We claim that f is neither injective nor surjective.

Pf: Find we consider injectivity. f is injective if and only if whenever $f(x) = f(y)$, $x = y$. However, $f(1) = 2 = f(-1)$, so f is not injective.

Now, f is surjective if and only if $f(\mathbb{Z}) = \mathbb{Z}_{\geq 0}$; in other words, if and only if for every $y \in \mathbb{Z}_{\geq 0}$, there exists $x \in \mathbb{Z}$ s.t. $f(x) = y$. We claim that $0 \notin f(\mathbb{Z})$. To see this, notice that for all $x \in \mathbb{Z}$, $f(x) = |x| + 1 \geq 0 + 1 = 1 > 0$. So

$0 \notin f(\mathbb{Z})$, and f is not surjective. $\square$

4) Name 3 of the 7 properties that algorithms should have, give a short description, and write some pseudocode which fails at only this property.

Properties: input, output, definiteness, correctness, finiteness, effectiveness, generality

Finiteness: For any input, the algorithm should produce the desired output after a finite number of steps

```
int-sqrt(pos integer x):      (find √x if √x ∈ ℤ;
    i := 0                          otherwise, return -1)
    while(i² ≠ x)
        i := i+1
    return i
```

Correctness: The algorithm should produce the correct output to the desired problem

```
int-sqrt(pos integer x):
    i := 0
    while(x² ≠ i)
        i := i+1
    return i
```

Effectiveness: It must be possible to perform each step exactly, and in a finite amount of time

int-sqrt (pos. integer x):

$A := \{(i, i^2) \mid i \in \mathbb{Z}_+\}$

If $(i, x) \in A$ for some $i \in \mathbb{Z}$

return i

Otherwise, return -1

---

5) Find and prove a simple big-Θ estimate for

$$f(x) = (7n^n + n2^n + 3^n)(n! + 3^n)$$

(Heuristic: Choose fastest-growing term of each factor)

Let $g(x) = n^n \cdot n!$ We claim that $f(x)$ is $\Theta(g(x))$.

Pf: We show that f is $O(g)$ and g is $O(f)$

Let $C = 18$ $k = 10$

Then $n! \geq 3^n$ for all $x > k$ (this needs pf)

and $n^n \geq n2^n$, $n^n \geq 3^n$ for all $x > k$ (this needs pf)

So for all $x > k$,

$$\begin{aligned} |f(x)| &= (7n^n + n2^n + 3^n)(n! + 3^n) \\ &\leq (7n^n + n^n + n^n)(n! + n!) \\ &= 18n^n \, n! = C|g(x)|, \end{aligned}$$

so f is $O(g)$.

(other direction: $C = k = 1$ works)

6) How many license plates can be made using either
a) three digits followed by three uppercase letters or
b) three uppercase letters followed by three digits

a) We have 10 choices for each of digits 1, 2, 3 and 26 choices for each of digits 4, 5, 6, so by the product rule, there are

$$10 \cdot 10 \cdot 10 \cdot 26 \cdot 26 \cdot 26 = 17576000$$ such license plates

b) We have 26 choices for each of digits 1, 2, 3 and 10 choices for each of digits 4, 5, 6, so by the product rule, there are

$$26 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10 = 17576000$$ such license plates

There is no overlap between the two cases, so in total there are

$$17576000 + 17576000 = 35152000$$ valid license plates