Assignment 1 – Discrete Review

Answer the questions in the boxes provided on the question sheets. If you run out of room for an answer, add a page to the end of the document.

Related Readings: http://pages.cs.wisc.edu/~hasti/cs240/readings/

Name: Zinnia Nie

Wisc id: 908 319 4044

Logic

1. Using a truth table, show the equivalence of the following statements.

(a) $P \vee (\neg P \wedge Q) \equiv P \vee Q$

Solution:

P	Q	P∨Q
T	T	T
T	F	T
F	T	T
F	F	F

P	Q	¬P	¬P∧Q	P∨(¬P∧Q)
T	T	F	F	T
T	F	F	F	T
F	T	T	T	T
F	F	T	F	F

(b) $\neg P \vee \neg Q \equiv \neg(P \wedge Q)$

Solution:

P	Q	¬P	¬Q	¬P∨¬Q
T	T	F	F	F
T	F	F	T	T
F	T	T	F	T
F	F	T	T	T

P	Q	P∧Q	¬(P∧Q)
T	T	T	F
T	F	F	T
F	T	F	T
F	F	F	T

(c) $\neg P \vee P \equiv \text{true}$

Solution:

P	$\neg P$	$\neg P \vee P$
T	F	T
F	T	T

(d) $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$

Solution:

P	Q	R	$Q \wedge R$	$P \vee (Q \wedge R)$
T	T	T	T	T
T	T	F	F	T
T	F	T	F	T
T	F	F	F	T
F	T	T	T	T
F	T	F	F	F
F	F	T	F	F
F	F	F	F	F

P	Q	R	$P \vee Q$	$P \vee R$	$(P \vee Q) \wedge (P \vee R)$
T	T	T	T	T	T
T	T	F	T	T	T
T	F	T	T	T	T
T	F	F	T	T	T
F	T	T	T	T	T
F	T	F	T	F	F
F	F	T	F	T	F
F	F	F	F	F	F

Sets

2. Based on the definitions of the sets A and B, calculate the following: $|A|$, $|B|$, $A \cup B$, $A \cap B$, $A \setminus B$, $B \setminus A$.

   (a) $A = \{1, 2, 6, 10\}$ and $B = \{2, 4, 9, 10\}$

   Solution: $|A| = 4$ $|B| = 4$

   $A \cup B = \{1, 2, 4, 6, 9, 10\}$

   $A \cap B = \{2, 10\}$

   $A \setminus B = \{1, 6\}$

   $B \setminus A = \{4, 9\}$

   (b) $A = \{x \mid x \in \mathbb{N}\}$ and $B = \{x \in \mathbb{N} \mid x \text{ is even}\}$

   Solution: $|A| = \infty$ $|B| = \infty$

   $A \cup B = \{x / x \in N\}$

   $A \cap B = \{x \in N \mid x \text{ is even}\}$

   $A \setminus B = \{x \in N \mid x \text{ is odd}\}$

   $B \setminus A = \emptyset$

Relations and Functions

3. For each of the following relations, indicate if it is reflexive, antireflexive, symmetric, antisymmetric, or transitive.

   (a) $\{(x, y) : x \leq y\}$

   Solution: reflexive, transitive, antisymmetric

   (b) $\{(x, y) : x > y\}$

   Solution: antireflexive, transitive, antisymmetric

(c) $\{(x, y) : x < y\}$

Solution: antireflexive, transitive, antisymmetric

(d) $\{(x, y) : x = y\}$

Solution: reflexive, symmetric, transitive, antisymmetric

4. For each of the following functions (assume that they are all $f : \mathbb{Z} \to \mathbb{Z}$), indicate if it is surjective (onto), injective (one-to-one), or bijective.

(a) $f(x) = x$

Solution: bijective

(b) $f(x) = 2x - 3$

Solution: bijective

(c) $f(x) = x^2$

Solution: None of the above

5. Show that $h(x) = g(f(x))$ is a bijection if $g(x)$ and $f(x)$ are bijections.

Solution: f(x) is a bijection so it is injective and surjective. f(x) is surjective so each input in the domain corresponds with an output value in the domain. g(x) is also onto, so it also outputs the entire domain and h(x) is thus surjective. f(x) and g(x) are injective, so a single input in f(x) has a single output which becomes input to g(x) which then has a single output. h(x) is thus injective and is a bijection.

Induction

6. Prove the following by induction.

(a) $\sum_{i=1}^{n} i = n(n+1)/2$

Solution: Base Case: $n=1$ $\quad \sum_{i=1}^{1} i = 1(1+1)/2$

$1 = 1$ ✓

Induction hypothesis: $\sum_{i=1}^{k} i = k(k+1)/2$ is true

→ Prove $n = k+1$ makes the expression true

$\sum_{i=1}^{k+1} i = ((k+1)(k+2))/2 \rightarrow \sum_{i=1}^{k} i + (k+1) = k(k+1)/2 + (k+1)$

induction hypothesis

$= \frac{(k^2+k)+2k+2}{2}$

$= \frac{k^2+3k+2}{2}$

equivalent

$= \frac{(k+1)(k+2)}{2}$ ∎

(b) $\sum_{i=1}^{n} i^2 = n(n+1)(2n+1)/6$

Solution: Base Case: $n=1$ $\quad \sum_{i=1}^{1} i^2 = 1(1+1)(2(1)+1)/6$

$1 = 1$ ✓

Induction hypothesis: $\sum_{i=1}^{k} i^2 = k(k+1)(2k+1)/6$ is true

→ Prove $n=k+1$ makes the expression true.

$\sum_{i=1}^{k+1} i^2 = ((k+1)(k+2)(2k+3))/6$

$\sum_{i=1}^{k} i^2 + (k+1)^2 = \frac{k(k+1)(2k+1)}{6} + (k+1)^2 = \frac{2k^3+3k^2+k}{6} + \frac{6k^2+12k+6}{6}$

induction hypothesis $= \frac{2k^3+9k^2+13k+6}{6} = \boxed{\frac{(k+1)(k+2)(2k+3)}{6}}$ ∎

(c) $\sum_{i=1}^{n} i^3 = n^2(n+1)^2/4$

Solution: Base Case: $n=1$ $\quad \sum_{i=1}^{1} i^3 = 1^2(1+1)^2/4$

$1 = 1$ ✓

Induction hypothesis: $\sum_{i=1}^{k} i^3 = k^2(k+1)^2/4$ is true

→ Prove $n=k+1$ makes the expression true.

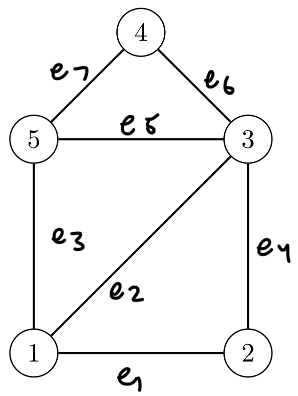
$\sum_{i=1}^{k+1} i^3 = (k+1)^2(k+2)^2/4$

$\sum_{i=1}^{k} i^3 + (k+1)^3 = \frac{k^2(k+1)^2}{4} + (k+1)^3 = \frac{k^4+2k^3+k^2}{4} + \frac{4k^3+12k^2+12k+4}{4}$

induction hypothesis $= \frac{k^4+6k^3+13k^2+12k+4}{4} = \boxed{\frac{(k+1)^2(k+2)^2}{4}}$ ∎

Graphs and Trees

7. Give the adjacency matrix, adjacency list, edge list, and incidence matrix for the following graph.



Solution:

$$A = \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}$$

1 → 2 → 3 → 5 X

2 → 1 → 3 X

3 → 1 → 2 → 4 → 5 X

4 → 3 → 5 X

5 → 1 → 3 → 4 X

Edge List: 1-2, 1-3, 1-5, 2-1, 2-3, 3-1, 3-2, 3-4, 3-5, 4-3, 4-5, 5-1, 5-3, 5-4

$$A_c = \begin{array}{c} \begin{array}{ccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}$$

8. How many edges are there is a complete graph of size n? Prove by induction.

Solution: Prove K_n has $\frac{n(n-1)}{2}$ edges

Base Case: $n=1$

A graph of size $n=1$ is just a single vertex with $\emptyset$ edges.

$\frac{1(1-1)}{2} = \emptyset$, so the equation works.

Induction Hypothesis: K_k has $\frac{k(k-1)}{2}$ edges

→ Prove $n=k+1$ makes the expression true

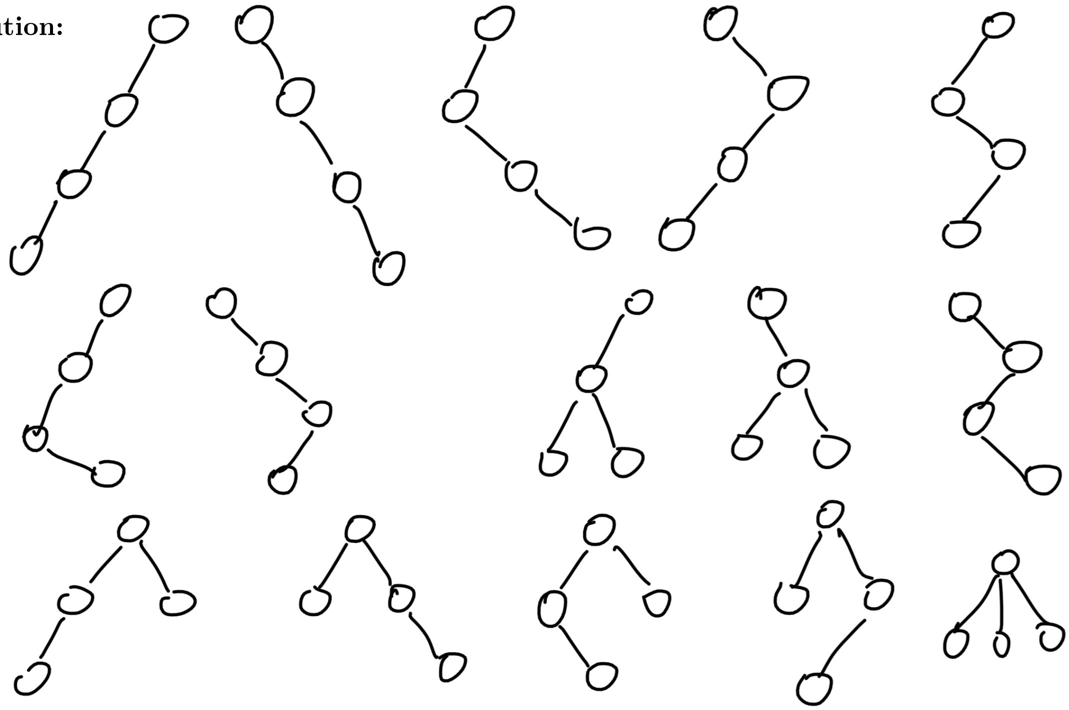
(K_{k+1} has $\frac{k(k+1)}{2}$ edges)

Every node in K_{k+1} has a degree of k. We know due to the induction hypothesis that K_k has $\frac{k(k-1)}{2}$ edges, and the $k+1$ node has degree k.

Thus K_{k+1} has $\frac{k(k-1)}{2} + k = \frac{k^2-k}{2} + \frac{2k}{2} = \frac{k^2+k}{2} = \boxed{\frac{k(k+1)}{2} \text{ edges}}$ ■

9. Draw all possible (unlabelled) trees with 4 nodes.

Solution:



10. Show by induction that, for all trees, $|E| = |V| - 1$.

Solution: Base Case: $|V| = 1$
$|E| = 1 - 1 = 0$

A one vertex graph has no edges.

Induction hypothesis: For $|V| = k$, $|E| = k - 1$

→ Prove when $|V| = k+1$, $|E| = k+1-1 = k$

In a tree, all leaf nodes have a degree of 1 and will add 1 edge to the total number of edges.

From the induction hypothesis, $|E| = k-1$ when $|V| = k$.

Then adding another vertex so $|V| = k+1$ would add an edge to the tree, making it so that $|E| = k$, which is the same as the result from the equation.

Counting

11. How many 3 digit pin codes are there?

Solution: $10 \cdot 10 \cdot 10 = 1000$

12. What is the expression for the sum of the ith line (indexing starts at 1) of the following:

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
⋮
16 17 18 19 20 21

Solution:

number of elements in nth row = n

1
5
15
34
65
111

starting value = $\frac{n}{2}(n-1)+1$

last value = $\frac{n}{2}(n-1)+n$

Gauss: pair values

$\hookrightarrow n(n-1)+1+n = n^2+1$

$\frac{n}{2}(n^2+1) = \boxed{\frac{n^3+n}{2}}$

13. A standard deck of 52 cards has 4 suits, and each suit has card number 1 (ace) to 10, a jack, a queen, and a king. A standard poker hand has 5 cards. For the following, how many ways can the described hand be drawn from a standard deck.

(a) A royal flush: all 5 cards have the same suit and are 10, jack, queen, king, ace.

Solution: 4 hands, one from each suit

(b) A straight flush: all 5 cards have the same suit and are in sequence, but not a royal flush.

Solution: 9 hands per suit $\rightarrow$ 36 straight flush hands

(c) A flush: all 5 cards have the same suit, but not a royal or straight flush.

Solution:
Possible flushes = $\binom{13}{5} = 5148$ $5148 - 40 = 5108$ flushes
40 royal / straight flushes

(d) Only one pair (2 of the 5 cards have the same number/rank, while the remaining 3 cards all have different numbers/ranks):

Solution: pick pair rank, pick pair suits, pick other ranks, pick suits

$\binom{13}{1} \cdot \binom{4}{2} \cdot \binom{12}{3} \cdot \binom{4}{1}^3 = 1098240$ hands

Proofs

14. Show that $2x$ is even for all $x \in \mathbb{N}$.
    (a) By direct proof.

Solution:

The def of even states that for every even number, $\exists k \in \mathbb{N}$ such that $n = 2k$.

$x \in \mathbb{N}$, so $2x$ is even by def of even. ■

    (b) By contradiction.

Solution: Assume $2x$ is odd.

This means, by def of odd, $\exists j \in \mathbb{N}$ such that

$2x = 2j+1$

$x = j + \frac{1}{2}$

$j \in \mathbb{N}$, but a natural number plus $\frac{1}{2}$ is not a natural number, so it is not possible for $2x$ to be odd, meaning $2x$ can only be even. ■

15. For all $x, y \in \mathbb{R}$, show that $|x+y| \le |x| + |y|$. (Hint: use proof by cases.)

Solution:

Case 1: both x and y are positive / negative

if they are both positive then the absolute value doesn't affect the sums and $x+y = x+y$.

if they are both negative, similarly

$|-x+-y| \le |-x| + |-y|$, $|-(x+y)| \le x+y$, $x+y = x+y$.

Thus the inequality is true

Case 2: one of x or y is negative and one is positive.

In this case, the right side will become $x+y$ due to the absolute values, however the left side will be valued at $|x-y|$.

$|x-y| < x+y$, so the inequality is true. ■

Program Correctness (and Invariants)

16. For the following algorithms, describe the loop invariant(s) and prove that they are sound and complete.

(a)

Algorithm 1: findMin

Input: a: A non-empty array of integers (indexed starting at 1)
Output: The smallest element in the array

```
begin
    min ← ∞
    for i ← 1 to len(a) do
        if a[i] < min then
            min ← a[i]
        end
    end
    return min
end
```

Solution: loop invariants: at the end of iteration i, min contains the minimum of the array $a[1 \ldots i]$

Base Case: $i = 1$
min contains ∞ in the first iteration, so min will be reassigned with $a[1]$ which is the min of $a[1 \ldots 1]$ as it is the only value.

Induction hypothesis: $i = k$, at the end of the kth iteration of the loop, min contains the minimum of the array $a[1 \ldots k]$.

Prove the invariant holds at the end of the $k+1$ iteration.
During that iteration min contains the minimum within $a[1 \ldots k]$ by the induction hypothesis. $a[i]$ is then $a[k+1]$. If $a[k+1] < min$ then min is updated, otherwise the min of $a[1 \ldots k]$ is the min of $a[1 \ldots k+1]$.
The loop invariant is true for all iterations. ∎

Termination: The array length is finite so the for loop will eventually terminate. Then min is returned which should contain the correct result. ∎

(b)

Algorithm 2: InsertionSort

Input: a: A non-empty array of integers (indexed starting at 1)
Output: a sorted from largest to smallest
begin
  for $i \leftarrow 2$ **to** $len(a)$ **do**
    $val \leftarrow a[i]$
    for $j \leftarrow 1$ **to** $i-1$ **do**
      if $val > a[j]$ **then**
        shift $a[j..i-1]$ to $a[j+1..i]$
        $a[j] \leftarrow val$
        break
      end
    end
  end
  return a
end

Solution: loop invariant: the array a[1... i] is sorted from largest to smallest at the end of the iteration

Base Case: in the internal for loop, a[1] and a[2]
i=2 are swapped if a[2] > a[1], so the resulting array a[1...2] is sorted.

Induction Hypothesis: i=k, the array a[1...k] is sorted from largest to smallest

Prove after the i=k+1 iteration the loop invariant holds.
The second for loop loops through the sorted portion of the array, which is a[1... i-1] = a[1...k], which is sorted by the induction hypothesis, until it finds a location, a[j], that is less than the unsorted value. This will placed the value in the sorted location because this part of the array is already sorted. The bottom half of the array from a[j ... i-1] is shifted one down to indices a[j+1... i]. Thus the array from a[1...i] is sorted and the loop invariant holds. $\blacksquare$

Termination: both for loops are bound with finite values so they will eventually terminate and the sorted array will be returned $\blacksquare$

Recurrences

17. Solve the following recurrences.

(a) $c_0 = 1; c_n = c_{n-1} + 4$

Solution:

$c_n = (c_{n-2} + 4) + 4 = ((c_{n-3} + 4) + 4) + 4$

$= c_{1-1} + 4 \ldots \ldots + 4$

$= 1 + 4 \ldots + 4$

$\boxed{c_n = 1 + 4n}$

(b) $d_0 = 4; d_n = 3 \cdot d_{n-1}$

Solution:

$d_n = 3 \cdot 3 \cdot d_{n-2} = 3 \cdot 3 \cdot 3 \cdot d_{n-3}$

$= 3 \cdot \ldots \cdot 3 \cdot d_0$

$= 3 \cdot \ldots \cdot 3 \cdot 4$

$\boxed{d_n = 4 \cdot 3^n}$

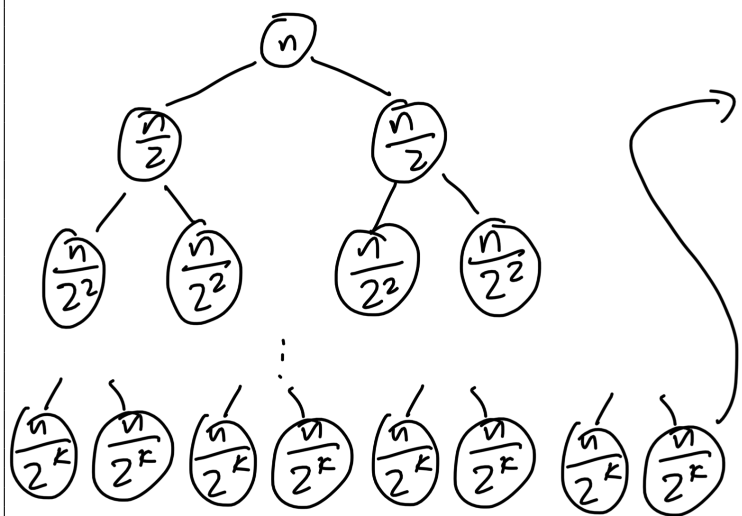
(c) $T(1) = 1; T(n) = 2T(n/2) + n$ (An upper bound is sufficient.)

Solution:

$$T(n) = 2\left(2T\left(\frac{n}{4}\right) + \frac{n}{2}\right) + n$$

$$= 2\left(2\left(2T\left(\frac{n}{2^3}\right) + \frac{n}{2^2}\right) + \frac{n}{2}\right) + n$$

$$\vdots$$



$$\sum_{i=0}^{k} 2^i \cdot \frac{n}{2^i}$$

$$= (k+1)n$$

$$= (\text{tree height} + 1)n$$

$$= (\text{tree height})n + n$$

$$= n\log(n) + n$$

$$= \boxed{O(n \log n)}$$

(d) $f(1) = 1; f(n) = \sum_1^{n-1}(i \cdot f(i))$
(Hint: compute $f(n+1) - f(n)$ for $n > 1$)

Solution:

$$f(n+1) - f(n) = \sum_1^{n}(i \cdot f(i)) - \sum_1^{n-1}(i \cdot f(i))$$

$$= \sum_1^{n-1}(i \cdot f(i)) + (n \cdot f(n))$$

$$- \sum_1^{n-1}(i \cdot f(i))$$

$$= n \cdot f(n)$$

$$f(n+1) - f(n) = n!$$

Not sure how to finish

Coding Question

Most assignments will have a coding question. You can code in C, C++, C#, Java, Python, or Rust. You will submit a Makefile and a source code file.

Makefile: In the Makefile, there needs to be a build command and a run command. Below is a sample Makefile for a C++ program. You will find this Makefile in assignment details. Download the sample Makefile and edit it for your chosen programming language and code.

```
#Build commands to copy:
#Replace g++ -o HelloWorld HelloWord.cpp below with the appropriate command.
#Java:
#       javac source_file.java
#Python:
#       echo "Nothing to compile."
#C#:
#       mcs -out:exec_name source_file.cs
#C:
#       gcc -o exec_name source_file.c
#C++:
#       g++ -o exec_name source_file.cpp
#Rust:
#       rustc source_file.rs

build:
        g++ -o HelloWorld HelloWord.cpp

#Run commands to copy:
#Replace ./HelloWorld below with the appropriate command.
#Java:
#       java source_file
#Python 3:
#       python3 source_file.py
#C#:
#       mono exec_name
#C/C++:
#       ./exec_name
#Rust:
#       ./source_file

run:
        ./HelloWorld
```

HelloWorld Program Details The input will start with a positive integer, giving the number of instances that follow. For each instance, there will be a string. For each string s, the program should output Hello, s! on its own line.

A sample input is the following:

```
3
World
Marc
Owen
```

The output for the sample input should be the following:

```
Hello, World!
Hello, Marc!
Hello, Owen!
```