CS 577 Assignment 1 – Discrete Review Spring 2023

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: ______________________ Wisc id: ______________________

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	$\neg P$	$\neg P \wedge Q$	$P \vee (\neg P \wedge Q$	$P \vee Q$
T	T	$\bot$	$\bot$	T	T
T	$\bot$	$\bot$	$\bot$	T	T
$\bot$	T	T	T	T	T
$\bot$	$\bot$	T	$\bot$	$\bot$	$\bot$

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

Solution:

P	Q	$\neg P$	$\neg Q$	$\neg P \vee \neg Q$	$(P \wedge Q)$	$\neg(P \wedge Q$
T	T	$\bot$	$\bot$	$\bot$	T	$\bot$
T	$\bot$	$\bot$	T	T	$\bot$	T
$\bot$	T	T	$\bot$	T	$\bot$	T
$\bot$	$\bot$	$\bot$	T	T	$\bot$	T

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

Solution:

P	$\neg P$	$\neg P \vee P$
T	⊥	T
⊥	T	T

TAUTOLOGY

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

Solution:

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

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: $|A| = \infty$ countable
$|B| = \infty$ countable
$A \cup B = \{x \mid x \in \mathbb{N}\} = A$
$A \cap B = \{x \in \mathbb{N}, x \text{ is even}\} = B$
$A \setminus B = \{x \in \mathbb{N}, x \text{ is odd}\} = B^c$
$B \setminus A = \emptyset$

Relations and Functions

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

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

Solution: reflexive: yes symmetric: no transitive: yes
antireflexive: no antisymmetric: no

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

Solution: reflexive: NO symmetric: NO transitive: yes
antireflexive: yes antisymmetric: yes

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

Solution: reflexive: no symmetric: no transitive: yes
antireflexive: yes antisymmetric: yes

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

Solution: reflexive: yes symmetric: yes transitive: yes
antireflexive: no anti sym: no

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: injective: yes surjective: yes bijective: yes

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

Solution: injective: yes surjective: yes bijective: yes

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

Solution: injective: no surjective: no bijective: no

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

Solution: h(x) is injective, so if h(x)=h(y), then x=y
h(x) is surjective, for every y in the codomain of h, there exists an x in the domain of h such that h(x)=y
Since g(x) and f(x) are bijections, we have:
g(x) is injective, if g(x)=g(y) then x=y
g(x) is onto, for every y in the codomain of g(x), there exists any x in the domain of g(x) such that g(x)=y
The same applies for f(x) (for injection and surjection)
If h(x)=h(y) then g(f(x)=g(f(y)). By one to one property of g(r), we have f(x)=f(y). By the one to one property on f(x) we have x=y, so h(x) is one to one.
For every y in the codomain of h(x), there exists a z in the codomain of g(x) such that y=g(z). By the onto property of f(x) there exists a ~~x~~ in the domain of f(x) such that f(x)=z.
Hence, h(x)=g(f(x)=g(z)=y so h is onto.

Induction

6. Prove the following by induction.

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

Solution: Base Case: $n=1 \Rightarrow \sum_{1}^{1} 1 = 1\cdot(1+1)/2 = 1$ ✓

Induction: let $n=k$ for which the statement $\sum_{i=1}^{k} i = k(k+1)/2$ is T.

We need to prove for $k+1 \Rightarrow$

$$\sum_{i=1}^{k+1} i = \sum_{i=1}^{k} i + (k+1) = \frac{k(k+1)}{2} + (k+1) = (k+1)\left(\frac{k}{2}+1\right) = \frac{(k+1)(k+2)}{2}$$

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

Solution: Base Case: $n=1 \Rightarrow \sum_{i=1}^{1} i^2 = \frac{1\cdot 2\cdot 3}{6} = 1$ ✓

Induction: let $n=k$ for which $\sum_{i=1}^{k} i^2 = k(k+1)(2k+1)/6$ is T.

We need to prove for $k+1 \Rightarrow$

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

$$= (k+1)\left(\frac{k\cdot(2k+1)}{6} + (k+1)\right) = (k+1)\frac{2k^2+7k+6}{6}$$

$$= \frac{(k+1)}{6}\cdot 2(k+2)\left(\frac{2k+3}{2}\right)$$

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

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

Induction: let $n=k$ for which $\sum_{i=1}^{k} i^3 = k^2(k+1)^2/4$ is T.

$\Rightarrow$ For $(k+1)$ we have:

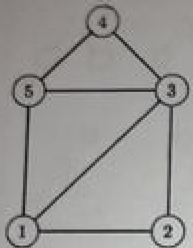
$$\sum_{i=1}^{k+1} i^3 = \sum_{i=1}^{k} i^3 + (k+1)^3 = \frac{k^2(k+1)^2}{4} + (k+1)^3 = (k+1)^2\left(\frac{k^2}{4} + (k+1)\right) =$$

$$= (k+1)^2\,\frac{k^2+4k+4}{4} = \frac{(k+1)^2\cdot(k+2)^2}{4}$$

Check: if $\sum_{i=1}^{k} i^3 = \frac{k^2(k+1)^2}{4}$ for $(k+1) \Rightarrow \sum_{i=1}^{k+1} i^3 = \frac{(k+1)^2(k+1+1)^2}{4}$ ✓

Graphs and Trees

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



Solution:

Adjacency Matrix

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

Adjacency List

1 → 2 | 5 | 3

2 → 1 | 3

3 → 1 | 2 | 4 | 5

4 → 3 | 5

5 → 1 | 3 | 4

Edge List: (1,2), (1,3), (1,5), (2,1), (2,3), (3,2), (3,1), (3,5), (3,4), (4,3), (4,5), (5,1), (5,3), (5,4)

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

Solution: We have the formula $\frac{n(n-1)}{2}$ for the maximum # of edges in a graph.

Base Case: graph with 1 node $\Rightarrow \frac{1 \cdot 0}{2} = 0$ edges T.

Induction: Let $S(k)$ for $k > 1$ be T. $\Rightarrow S(k) = \frac{k(k-1)}{2}$ is T.

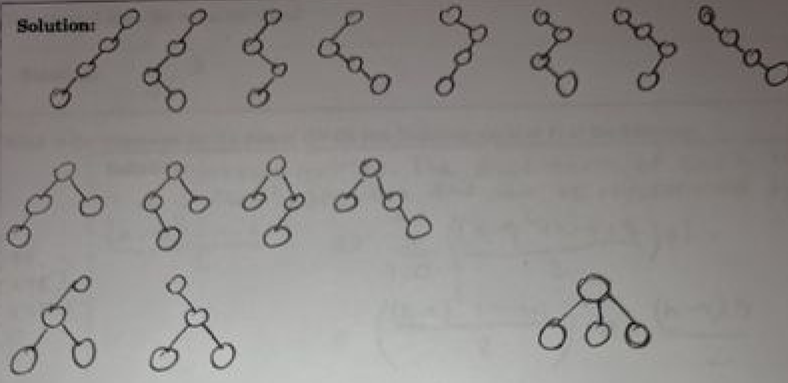
$$S(k+1) = S(k) + k = \frac{k(k-1)}{2} + k = k\left(\frac{k-1}{2} + 1\right) =$$

$$= k \cdot \frac{k-1+2}{2} = \frac{k(k+1)}{2}$$

$$\rightarrow \frac{(k+1)(k+1-1)}{2} = \frac{(k+1)k}{2} \quad \checkmark$$

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

Solution:



Total: 15

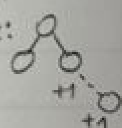
10. Show by induction that, for all trees, $|E| = |V| - 1$.

Solution: Base Case $|V| = 1 \Rightarrow |E| = 0 \Rightarrow T$

Induction: Let $|V| = |K|$ for which $|E| = |K| - 1$ is T

When we have a tree, if we add one vertex, we can gain maximum one edge.

Ex:



$\Rightarrow |V| = |K+1| \Rightarrow |E|+1 = |K|+1-1 \Rightarrow |E|+1 = |K|$

$\Rightarrow |E| = |K| - 1$

Counting

11. How many 3 digit pin codes are there?

Solution: 10^3

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
⋮

Solution: n = raw number. The first term of each row is a caterer's sequena and can be represented by $\frac{(n-1)^2+n-1+2}{2}$ $\Rightarrow \sum_{i=0}^{n-1}\left(\frac{(n-1)^2+n-1+2}{2}\right)+i =$

$$= \left(\frac{(n-1)^2+n-1+2}{2}\right)\cdot n + \frac{(n-1)n}{2}$$

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

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

Solution: $9 \cdot 4 = 36$

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

Solution: $\binom{13}{5}\cdot 4 - 36 - 4 = 1287 \cdot 4 - 40 = 5108$

(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: $\binom{4}{2}\cdot 13 \cdot \binom{48}{1}\binom{44}{1}\binom{40}{1} / 3!$

Proofs

14. Show that $2x$ is even for all $x \in \mathbb{N}$.

(a) By direct proof.

Solution: a) Let $x=2k$

$\Rightarrow 2x = 2(2k) \Rightarrow$ even

b) Let $x = 2k+1$

$\Rightarrow 2x = 2(2k+1) = 4k+2 \Rightarrow$ even

Every number multiplied by two is an even num.

(b) By contradiction.

Solution: Let $2x$ is odd. $x \in \mathbb{N}$

$\Rightarrow 2x = 2k+1$ $k \in \mathbb{N}$

$2k = 2x-1$

$k = \frac{2x-1}{2} = x - \frac{1}{2}$ we gets not a natural number

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

Solution: 1. X any y are positive. $\Rightarrow |x+y| = x+y = |x|+|y|$ because $x,y > 0 \Rightarrow |x+y| \leq |x|+|y|$

2. X any Y are negative. Therefore $x+y = -|x+y|$ because $x,y<0$. Hence $-|x+y| = -(|x|+|y|) \Rightarrow |x+y| = |x|+|y| \Rightarrow |x+y| \leq |x|+|y|$

3. $x < 0 < y$

a) $|x| < |y|$

$|x+y| = x+y < |x|+|y|$

$\Rightarrow |x+y| \leq |x|+|y|$

b) $|x| > |y|$

$|x+y| = -(x+y) < |x|+|y|$

$\Rightarrow |x+y| \leq |x|+|y|$

Program Correctness (and Invariants)

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

Algorithm 1: findMin

(a)

```
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 Invariant: At the start of the iteration with index i of the loop, the variable min should contain the minimum of the numbers a[1...i]
*Initialization: at the start of the first loop we have i = 1. min should contain the minimum of the numbers from the subarray a[1.. 1] which is a[1]
Maintenance: Assume loop invariant min at the start of iteration i. contains minimum of numbers in subarray a[1...i]. There are two cases:
1. a[i] < min. From the loop invariant we get that a[i] is smaller than the minimum of the numbers. Thus a[i] is the minimum of that subarray. In this case the Algorithm sets min to a[i], thus in this case the loop invariant holds again at the beginning of the next loop
2. a[i] ≥ min. That is the minimum in a[1..i] is at least as large as a[i], thus the minimum in the array[1..i] remains the same. The algorithm also doesn't change answer, thus the loop invariant holds again at the beginning of the new iteration
Termination: When the for loop terminates i = n. Now the loop invariant gives: the variable min contains the minimum of all numbers in the subarray a[1...N]

*Before the loop min = ∞ *min is Loop invariant

Proof of Completeness: Since the loop iterates over all the elements in the array a, the value of min will be updated if and only

if there exists an element in the array a that is smaller than min. This Ensures that the algorithm will find the smallest element in an array "a", with finite number of steps

(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 ← 2 to len(a) do
        val ← a[i]
        for j ← 1 to i − 1 do
            if val > a[j] then
                shift a[j..i − 1] to a[j + 1..i]
                a[j] ← val
                break
            end
        end
    end
    return a
end
```

Solution: loop invariant: At the start of each iteration of the outer loop, the sub-array "a[1..i-1]" is sorted in decreasing order.
Proof of soundness: 1. Initialization: Before the start of the outer loop, the sub-array a[1..1] is considered as sorted.
2. Maintenance: At the start of each iteration of the outer loop the subarray a[1..i-1] is sorted in decreasing order by the loop invariant. The inner loop inserts the value "Val" at its proper place in the sub-array a[1..i-1] by shifting the elements to the right if necessary and breaking when the proper place is found This ensures that the sub array a[1..i] remains sorted in decreasing order
3. Termination: After the outer loop terminates, the sub-array a[1..len(a)] is sorted in decreasing order, which is the entire array a. So the array a is sorted from largest to smallest
Proof of Completeness: Since the inner loop will insert the value Val at its proper place in the sub-array a[1..i-1] by shifting the ~~elemen~~ elements to the right if necessary, the algorithm ensures that the subarray a[1..i] remains sorted. The outer loop will iterate through all the elements in the array 'a' so the array a will be completely sorted in the end.
Therefore, the loop invariant is both sound and complete.

Recurrences

17. Solve the following recurrences.

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

Solution:
$C_0 = 1$
$C_1 = C_0 + 4 = 1+4 = 5$
$C_2 = C_1 + 4 = (C_0 + 4) + 4$
$C_3 = C_2 + 4 = (C_1 + 4) + 4 = \cdots$
$\vdots$
$C_n = C_{n-1} + 4 = (C_{n-2} + 4) + 4 = \cdots$
$C_n = (((C_0 + \underbrace{4) + 4) + 4) + \cdots + 4}_{n}$

$\boxed{C_n = C_{n-1} + 4n}$

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

Solution:
$d_0 = 4$
$d_1 = 3d_0 = 3 \cdot 4$
$d_2 = 3d_1 = 3 \cdot 3d_0$
$\vdots$
$d_n = 3d_{n-1} = 3(3d_{n-2}) \cdots$
$d_n = ((d_0 \cdot \underbrace{3) \cdot 3) \cdot 3) \cdot \cdots \cdot 3}_{n}$

$\Rightarrow \boxed{d_n = 3^n d_0 = 3^n \cdot 4}$

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

Solution: $T(n/2) = 2T(n/2^2) + \frac{n}{2}$

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

$\vdots$

$T(n) = 2^3 T(n/2^3) + 3n$

$\frac{n}{2^k} = 1$

$\vdots$

$n = 2^k$

$T(n) = 2^k T(n/2^k) + kn$

$k = \log n$

$T(n) = 2^{\log n} T(1) + n\log n$

$T(n) = n + n\log n = n(1 + \log n) \approx n\log n$

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

Solution: $f(n+1) = f(n) + nf(n)$

$f(n+1) = f(n) \cdot (n+1) = (n+1)f(n)$

$n = 1 \Rightarrow f(2) = 2f(1)$

$n = 2 \Rightarrow f(3) = 3 \cdot (2f(1))$

$\Rightarrow f(n+1) = (n+1)! \cdot 1 = \boxed{f(n) = n!}$