CS 336

1. The important issue is the logic you used to arrive at your answer.
2. Use extra paper to determine your solutions then neatly transcribe them onto these sheets.
3. Do not submit the scratch sheets. However, all of the logic necessary to obtain the solution should be on these sheets.
4. Comment on all logical flaws and omissions and enclose the
comments in boxes

1. [10] Let A and B be non-empty sets with cardinalities m and n, respectively. How many functions from A to B are **not** one-to-one? (You may assume $m \le n$.)

For each of the m elements of A there are n choices for the value of the function so there are n^m functions from A to B. To count the one-to-one functions, we notice that there are There are n choices for the value of the function for the first element of A, $n-1$ choices for the value of the function for the first element of A, ..., and $n-m+1$ choices for the value of the function for the m^{th} element of A. Thus, there are $\frac{n!}{(n-m)!}$ one-to-one functions from A to B. There are $n^m - \frac{n!}{(n-m)!}$ functions from A to B that are not one-to-one.

2. Consider choosing k objects from a set $\{a_1, a_2, \ldots, a_n\}$ of n objects with order unimportant but repetition allowed.

a. **[5]** How many such selections have exactly one a_1 ?

We need to determine how many ways we have of choosing $k-1$ objects from a set of the set $\{a_2, \ldots, a_n\}$ of $n-1$ objects with order unimportant but repetition allowed. This is equivalent to placing $k-1$ balls in $n-1$ bins and there are

$$\binom{(n-1)+(k-1)-1}{k-1} = \binom{n+k-3}{k-1}$$

ways of doing it.

[5] How many such selections have at least a_1 ?

Since the total number of ways of choosing k objects from a set $\{a_1, a_2, \ldots, a_n\}$ of n objects with order unimportant but repetition allowed is $\binom{n+k-1}{k}$ and there are $\binom{(n-1)+k-1}{k} = \binom{n+k-2}{k}$ ways of choosing k objects from the set $\{a_2, \ldots, a_n\}$ of n objects with order unimportant but repetition allowed, there are $\binom{n+k-1}{k} - \binom{n+k-2}{k}$ such selections having at least one a_1. Alternatively, you could see this as choosing a_1 once then choosing $k-1$ objects from the set $\{a_1, a_2, \ldots, a_n\}$. This way you get $\binom{n+k-2}{k-1}$ which equals $\binom{n+k-1}{k} - \binom{n+k-2}{k}$.

3. a. [10] Using a combinatorial argument, prove that for $n \geq 1$:

$$3^n = \sum_{k=0}^{n} \binom{n}{k} 2^k$$

We seek to determine how many strings of length n there are consisting of elements of $\{a, b, c\}$ allowing repetition. Since there are three choices for each of the n positions there are 3^n such strings. Alternatively, let k denote the number of positions in the string occupied by a or b. The value of k varies from 0 to n. For a fixed value of k, there are $\binom{n}{k}$ ways to select these positions and then 2 options for each of the k positions, thus $\binom{n}{k}2^k$ for the fixed value of k and $\sum_{k=0}^{n} \binom{n}{k} 2^k$ overall. This must equal 3^n.

b. [10] Using a combinatorial argument, prove that for $m, n, p \geq 1$:

$$\binom{m+n+p}{m}\binom{n+p}{n} = \binom{m+n+p}{p}\binom{m+n}{n}.$$

Given a set S of cardinality $m+n+p$, consider how many partitions there are of S into disjoint subsets A, B, and C of cardinalities m, n, and p, respectively. For the left side of the equality, we count this by first selecting the m elements for the subset A in $\binom{m+n+p}{m}$ ways and then selecting the n elements for the subset B from the remainder in $\binom{n+p}{n}$ ways. The remaining elements form the subset C. For the right side of the equality, we count this by first selecting the p elements for the subset C in $\binom{m+n+p}{p}$ ways and then selecting the n elements for the subset B from the remainder in $\binom{m+n}{n}$ ways. The remaining elements form the subset A. The two must be equal so

$$\binom{m+n+p}{m}\binom{n+p}{n} = \binom{m+n+p}{p}\binom{m+n}{n}.$$

4. a. [5] Suppose k objects are being chosen from a set $\{a_1, a_2, \ldots, a_n\}$ of n objects with order **unimportant** and repetition **not** allowed. Suppose all such selections are equally likely. What is the probability that a selection contains **exactly two** of $\{a_1, a_2, a_3\}$? (You may assume $n \ge k+1 \ge 3$.)

There are $\binom{n}{k}$ equally likely selections of k objects from n objects with order unimportant and repetition not allowed. If exactly two of $\{a_1, a_2, a_3\}$ are chosen there are $\binom{3}{2}$ ways to select the two from $\{a_1, a_2, a_3\}$ and then $\binom{n-3}{k-2}$ ways to choose the remaining $k-2$ from the $n-3$ elements $\{a_4, a_5, \ldots, a_n\}$. So there are $\binom{3}{2}\binom{n-3}{k-2}$ selections containing exactly two of $\{a_1, a_2, a_3\}$ and the probability of such a selection is $\dfrac{\binom{3}{2}\binom{n-3}{k-2}}{\binom{n}{k}}$

b. [5] Now suppose k objects are being chosen from a set $\{a_1, a_2, \ldots, a_n\}$ of n objects with order **important** and repetition still **not** allowed. Suppose all such selections are equally likely. What is the probability that a selection contains **exactly two** of $\{a_1, a_2, a_3\}$?

There are $\dfrac{n!}{(n-k)!}$ equally likely selections of k objects from n objects with order important and repetition not allowed. If exactly two of $\{a_1, a_2, a_3\}$ are chosen there are $\binom{3}{2}$ ways to select the two from $\{a_1, a_2, a_3\}$ and then $\binom{n-3}{k-2}$ ways to choose the remaining $k-2$ from the $n-3$ elements $\{a_4, a_5, \ldots, a_n\}$. Then there are $k!$ ways to permute the elements selected. Thus, there are $\binom{3}{2}\binom{n-3}{k-2}k!$ selections containing exactly two of $\{a_1, a_2, a_3\}$ and the probability of such a selection is $\dfrac{\binom{3}{2}\binom{n-3}{k-2}k!}{\frac{n!}{(n-k)!}}$. (Alternatively, we could recognize that this is the same as part a. since, if orser is imposed, there are $k!$ ways of permuting the selections with exactly two of $\{a_1, a_2, a_3\}$ but also $k!$ ways of permuting all of the selections.)

5. [10] Using Definition 2' but no cardinality theorems, prove that the set

$$P = \{\textit{infinitely long strings of } 0s \textit{ and } 1s \textit{ with exactly two } 1s\}$$

is infinite.

Consider the function $f: P \to P$ defined by $f(s) = 0 \| s$ for any string $s \in P$ (where $\|$ denotes concatenation). Notice that if s has exactly two 1s, then so will $f(s)$. The function f is one-to-one since if $s, t \in P$ and $s \neq t$, then $f(s) = 0 \| s \neq 0 \| t = f(t)$. Lastly, notice that since no string maps to $< 110000... >$, f maps P into a proper subset of P. We conclude that P is infinite.

6. [10] Let A be a countably infinite set , B be an uncountably infinite set nonempty set and $C = A \times B$. Is the C finite, countably infinite, or uncountably infinite? Prove your assertion.

The set C is uncountably infinite. Since A is countably infinite, it is non-empty . Let a be any element of A. Consider the function $f: B \to C$ defined as $f(b) = (a, b)$. The function f is one-to-one since if $b_1, b_2 \in B$ and $b_1 \neq b_2$, then $f(b_1) = (a, b_1) \neq (a, b_2) = f(b_2)$. By Theorem 11, C is uncountably infinite.

7. [10] Given that for $n \geq 1$ and $\alpha > 0, (1+\alpha)^n \geq 1 + n\alpha$, prove that $n \cdot 2^n = \mathrm{O}(3^n)$.

Let $M = \frac{1}{2}$ and $N = 1$. For $n \geq 1$, $(\frac{3}{2})^n = (1 + \frac{1}{2})^n \geq 1 + \frac{n}{2} \geq \frac{n}{2}$. Thus, for $n \geq 1$, $2 \cdot 3^n \geq n \cdot 2^n$. So then for $n \geq N, |n \cdot 2^n| = n \cdot 2^n \leq 2 \cdot 3^n = M|3^n|$.

8. [10] Prove that if $f_1 = \mathrm{O}(g_1)$ and $f_2 = o(g_2)$, then $f_1 f_2 = o(g_1 g_2)$.

Since $f_1 = \mathrm{O}(g_1)$, there exist M and N_1 so that for $n \geq N_1, |f_1(n)| \leq M|g_1(n)|$.

Since $f_2 = o(g_2)$, given $\varepsilon > 0$ then also $\frac{\varepsilon}{M} > 0$ and there exists N_2 so that for

$n \geq N_2, |f_2(n)| \leq \frac{\varepsilon}{M}|g_2(n)|$, thus for

$n \geq \max\{N_1, N_2\}, |f_1(n) f_2(n)| \leq M|g_1(n)| \frac{\varepsilon}{M}|g_2(n)| = \varepsilon|g_1(n) g_2(n)|$, so

$f_1 f_2 = o(g_1 g_2)$.

9. [10] Assuming x and y are integer variables, prove correct with respect to precondition "y is defined" and postcondition "$x > 0$":

```
if y > 0 then
        x := 2*y
        if x > 5 then
                x : = x-4
        endif
else
        x := 4-y
        y := y-1
        if y = -3 then
                x : = x-3
        endif
endif
```

	y is defined
if y > 0 **then**	$y > 0$
x := 2*y	$y > 0 \wedge x = 2y$
	$x > 0$
if x > 5 **then**	$x > 11$
x : = x-4	$(x' > 5) \wedge (x = x' - 4)$
	$x > 1$
endif	$(x > 0) \vee (x > 1)$
	$x > 0$
else	$y \leq 0$
x := 4-y	$(y \leq 0) \wedge (x = 4 - y)$
	$x > 3$
y := y-1	$x > 3$
if y = -3 **then**	$x > 3$
x : = x-3	$(x' > 3) \wedge (x = x' - 3)$
	$x > 0$
endif	$(x > 3) \vee (x > 0)$
	$x > 0$
endif	$(x > 0) \vee (x > 0)$
	$x > 0$

10. [10] Prove the following code is partially correct with respect to precondition "*true*" and postcondition "*k is even*" . (Assume k, n, and i are integer variables and that n and i are defined at input.)

```
k := 1234
while i ≤= n do
    if i ≥ 7 then
        k : = k-12
    else
        k := 4*k-6
    endif
    i := i+5
endwhile
```

Be explicit about your loop invariant: I = "*k is even*"

```
______________________________ true
k := 1234______________________ k = 1234
______________________________ k is even
while i ≤= n do_____________ (k is even) ∧ (i ≤ n)
                _____________ k is even
    if i ≥ 7 then________ (k is even) ∧ (i ≥ 7)
              _____________ k is even
              k : = k-12____ (k' is even) ∧ (k = k' − 12)
              _____________ k is even
    else _______________ (k is even) ∧ (i < 7)
              _____________ k is even
              k := 4*k-6____ (k' is even) ∧ (k = 4k' − 6)
              _____________ k is even
    endif  _____________ (k is even) ∨ (k is even)
              _____________ k is even
    i := i+5_____________ (k is even) ∧ (i' = i + 5)
              _____________ k is even
endwhile_________________ (k is even) ∧ (i > n)
_____________________________ k is even
```

11. [10] Prove that the code below terminates. (Assume s and i are integer variables.):

```
s := 0
i := 1
while i ≤= 100000 do
        s := s+i
        i := 4*i+2
endwhile
```

First we recognize that if the quantity *100,000-i* becomes negative, the loop will terminate. We will show that that quantity strictly decreases but to that end we need to guarantee that the variable i stays positive. Consider the invariant "$i \geq 1$":

```
s := 0
i := 1____________________        i ≥ 1
while i ≤= 100000 do_____         i ≥ 1
        s := s+i ____________     i ≥ 1
        i := 4*i+2___________     (i' ≥ 1) ∧ (i = 4i' + 2)
              _______________     i = i' + 3i' + 2 > i' ≥ 1
              _______________     (i ≥ 1) ∧ (100000 − i < 100000 − i')
              _______________     i ≥ 1
endwhile
```

The quantity *100,000-i* strictly decreases through the loop. Since this is an integer expression, eventually *100,000-i* becomes negative and the loop terminates.

12. [10] Determine the weakest precondition with respect to the postcondition "$S = 0$" for the following (assume S, y, and x are integer variables and y and x are defined

```
if x≠0 then
        x := y
        S := x-y
else
        S := y+x
endif
```

$wp($ **if** x≠0 **then** x := y; S := x-y **else** S := y+x **endif,** $S = 0)$

$= ((x \neq 0) \wedge wp($ x := y; S := x-y, $S = 0)) \vee ((x = 0) \wedge wp($ S := y+x, $S = 0))$

$= ((x \neq 0) \wedge wp($ x := y; $x - y = 0)) \vee ((x = 0) \wedge (y + x = 0))$

$= ((x \neq 0) \wedge (y - y = 0)) \vee (x = y = 0))$

$= (x \neq 0) \vee (x = y = 0))$

13. [10] Determine the weakest precondition with respect to the postcondition "$x \neq y$" for the following (assume y and x are integer variables and x is defined). Simplify your answer so that there are **NO** logical operators.

```
if x ≥ 3 then
        y:= 2
else
        if x = 2 then
                y := 6
        else
                y := x+1
        endif
endif
```

We consider the inner if-then-else first:

wp (**if** $x = 2$ **then** y := 6 **else** y := x+1 **endif,** $x \neq y$)

$= ((x = 2) \wedge wp(\text{y := 6}, \ x \neq y)) \vee ((x \neq 2) \wedge wp(\text{ y := x+1}, \ x \neq y))$

$= ((x = 2) \wedge (x \neq 6)) \vee ((x \neq 2) \wedge (x \neq x + 1))$

$= ((x = 2) \vee (x \neq 2)$

$= true$

Now, letting S denote "**if** $x = 2$ **then** y := 6 **else** y := x+1 **endif**",

wp (**if** $x \geq 3$ **then** y:= 2 **else** S **endif,** $x \neq y$)

$= ((x \geq 3) \wedge wp(\text{y := 2}, \ x \neq y)) \vee ((x < 3) \wedge wp(\ S, \ x \neq y))$

$= ((x \geq 3) \wedge (x \neq 2)) \vee ((x < 3) \wedge true)$

$= ((x \geq 3) \vee (x < 3)$

$= true$

So, wp (**if** $x \geq 3$ **then** y:= 2 **else if** $x = 2$ **then** y := 6 **else** y := x+1 **endif** **endif,** $x \neq y$) = $true$.