Final Examination Solutions

1. [10] For $n \geq 7$, how many subsets of size 7 from $\{a_1, a_2, \ldots, a_n\}$ are there that either contain both a_1 and a_2 or both a_3 and a_4 (or all four)?

There are $\binom{n-2}{5}$ subsets of size 7 from $\{a_1, a_2, \ldots, a_n\}$ that contain both a_1 and a_2 since there are only 5 elements to choose from the remaining $n-2$. Similarly there are $\binom{n-2}{5}$ subsets of size 7 that contain both a_3 and a_4. There are $\binom{n-4}{3}$ subsets that contain all four, therefore there are $2\binom{n-2}{5} - \binom{n-4}{3}$ subsets that either contain both a_1 and a_2 or both a_3 and a_4 (or all four)

2. [10] Given $n \geq r \geq 1$, in how many ways can n identical balls be placed into r distinct bins such that each bin contains at least one ball? (Hint: Consider dot diagrams.)

Consider arrays of the form: $1 \quad 2 \quad 3 \quad \ldots r$
with n dots $\bullet$ interspersed in such a manner that at least one dot follows each number and no dot precedes 1. The number dots correspond to the number of balls in each bin. Since one dot must follow each number in the array, it must look like

$$1\bullet \quad 2\bullet \quad 3\bullet \quad \ldots r\bullet$$

The leading symbol pair "$1\bullet$" is fixed. There remain $r-1$ symbol pairs (number followed by dot) and $n-r$ dots to be distributed in the $(r-1)+(n-r) = n-1$ positions. Thus, there are $\binom{n-1}{n-r} = \binom{n-1}{r-1}$ positions for the dots.

3. a. [10] Using a combinatorial argument, prove that for $n \geq 2$ and $m \geq 2$:

$$\binom{n+m}{2} = n \cdot m + \binom{n}{2} + \binom{m}{2}$$

Consider subsets of two elements from the union of disjoint subsets A and B with cardinalities n and m, respectively. Since $\#(A \cup B) = n + m$, there are $\binom{n+m}{2}$ subsets of size two. Alternatively, consider that either one element comes from each of A and B, both from A, or both from B. These can be done in $n \cdot m, \binom{n}{2}$, and $\binom{m}{2}$ ways, respectively, and the total is $n \cdot m + \binom{n}{2} + \binom{m}{2}$. We conclude that

$$\binom{n+m}{2} = n \cdot m + \binom{n}{2} + \binom{m}{2}$$

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

$$\sum_{k=1}^{n} k\binom{n}{k} = n2^{n-1}$$

(Hint: Let *A* be a set of cardinality *n*. Consider pairs <*B, a*> where $B \subseteq A$ and $a \in A \sim B$.)

Employing the notation from the hint, and considering the left side of the equation first, there are n choices for a and then 2^{n-1} subsets from the remaining $n-1$ elements. Alternatively, let k be the number of elements in $\{a\} \cup B$. The value of k could range from 1 through n. For a fixed value of k, there are $\binom{n}{k}$ ways to choose $\{a\} \cup B$, and then k choices from this for a (with the remaining chosen elements forming B). There are $\sum_{k=1}^{n} k\binom{n}{k}$ total ways of doing this and this must equal $n2^{n-1}$.

4. a. [10] For $n \geq 5$, consider strings of length n using elements of $\{1,2,3\}$. Assume all such strings are equally likely. What is the probability that a string has exactly two 3's?

There are 3^n equally likely strings of length n using elements of $\{1,2,3\}$. However, if there are exactly two 3's, then there are $\binom{n}{2}$ ways to choose the positions for the two 3's and two options for each of the other $n-2$ positions yielding $\binom{n}{2}2^{n-2}$ possibilities. The probability of such a string thus is $\binom{n}{2}2^{n-2}/3^n$.

b. [5] What is the probability that such a string has exactly three 2's given that it has exactly two 3's?

There are $\binom{n}{2}$ ways to chose the positions for the two 3's, $\binom{n-2}{3}$ ways to chose the positions for the three 2's, and just one options for each of the other $n-5$ positions yielding $\binom{n}{2}\binom{n-2}{3}$ possibilities. The probability of having exactly three 2's and exactly two 3's is then $\binom{n}{2}\binom{n-2}{3}/3^n$, so the probability that such a string has exactly three 2's **given** that it has exactly two 3's is $\dfrac{\binom{n}{2}\binom{n-2}{3}/3^n}{\binom{n}{2}2^{n-2}/3^n} = \binom{n-2}{3}/2^{n-2}$.

5. **[10]** Using definition 2' (and no cardinality theorems) prove that the set of all integral multiples of 17 (i.e., $\{17k \mid k \in Z\}$ is infinite.

Let $A = \{17k \mid k \in Z\}$ and consider $f : A \to A$ defined by $f(n) = 2n$. This does map A into A since if *n* is an integral multiple of 17 so is *2n*. Furthermore, it is one-to-one since for $i \neq j$, $f(i) = 2 * i \neq 2 * j = f(j)$. Lastly, f maps A into a proper subset of itself since for no integer n is $2n = 17$ and yet $17 \in A$. Dy definition 2', A is infinte.

6. [10] Let $S = \{f \mid f : N \to \{0,1\}\}$ (i.e., the set of functions mapping the natural numbers into $\{0,1\}$). Is S finite, countably infinite, or uncountably infinite? State then prove your assertion.

S is uncountably infinite. Consider the mapping g from [0, 1] into S defined in the following manner: for $x \in [0,1]$, express x in binary as $x = .b_0 b_1 b_2 \ldots$ (terminating in an infinite string of 1's if there is an option between an infinite string of 0's or an infinite string of 1's). Define $f = g(x)$ as $f(n) = b_n$ (thus for $x = .011010\ldots$, $f = g(x)$ maps 0 to 0, 1 to 1, 2 to 1, 3 to 0, etc.). The function g is one-to-one since for $x \neq y$, the binary representation of x must differ form that for y in some digit k. So with $x = .b_0 b_1 b_2 \ldots$ and $y = .c_0 c_1 c_2 \ldots$, we have $b_k \neq c_k$. But then $g(x)$ maps k to b_k and $g(y)$ maps k to c_k, so $g(x) \neq g(y)$. Since g maps the uncountably infinite set [0, 1] one-to-one into S, by Theorem 9, S is uncounatbly infinite. [Note: this could also be proved with a diagonalization argument.]

7. [10] Prove that $n^3 + n^2 + n = o(n^4)$.

For $\varepsilon > 0$, let $N = \lceil 3/\varepsilon \rceil + 1$. For $n \geq N$, we have $n > 1$ and $n > 3/\varepsilon$, so $\varepsilon n > 3$ and $|n^3 + n^2 + n| = n^3 + n^2 + n < n^3 + n^3 + n^3 = 3n^3 < \varepsilon n \cdot n^3 = \varepsilon |n^4|$

8. a. [10] .Recalling that $\log_a x = \log_b x \cdot \log_a b$, prove that for all $a, b > 1$, if $f = O(\log_a n)$ then $f = O(\log_b n)$.

Since $f = O(\log_a n)$, there exist M and N so that if $n \geq N$, $|f(n)| \leq M |\log_a n|$. But then, if $n \geq N$, $|f(n)| \leq M |\log_a n| = M |\log_a b| |\log_b n| = M' |\log_b n|$, for $M' = M |\log_a b|$.

b. [10]. Consider the following assertion:

For all $a, b > 1$, if $f = O(a^n)$ then $f = O(b^n)$.

Using a simple example, prove that this assertion is false. Notice your example function must be $O(a^n)$ but not $O(b^n)$.

Let $a = 4$ and $b = 2$. Clearly $f(n) = 4^n$ is $O(4^n)$, since for $n \geq 0$, $|f(n)| = 4^n \leq 1 \cdot |4^n|$. But if $f = O(2^n)$, then there would have to exist M and N so that if $n \geq N$, $|f(n)| = 4^n \leq M \cdot 2^n$. Let $n = \max\{N, |\log_2 M| + 1\}$. We have $n \geq N$ and also $n > |\log_2 M| \geq \log_2 M$, so $2^n > M$. We conclude that $|f(n)| = 4^n = 2^n \cdot 2^n > M \cdot 2^n = M \cdot |2^n|$. We conclude that f is $O(4^n)$ but is not $O(2^n)$.

9. **[10]** Prove the following code is partially correct with respect to precondition "$n \geq 1$" and postcondition "$m = \max\{a_1, a_2, \ldots, a_n\}$":

```
m :=a[1]
i := 2
while i≤n do
        if m<a[i] then
                m := a[i]
        endif
        i := i+1
endwhile
```

You may assume that m, i, n, and the array a are integer variables. You may also assume that the array components $a_1, a_2, \ldots a_n$ are defined. Finally you may use the following axioms:

$$a_i \leq \max\{a_1, \ldots, a_{i-1}\} \Rightarrow \max\{a_1, \ldots, a_i\} = \max\{a_1, \ldots, a_{i-1}\})$$

$$a_i > \max\{a_1, \ldots, a_{i-1}\} \Rightarrow \max\{a_1, \ldots, a_i\} = a_i$$

____________________________ $n \geq 1$

m :=a[1]

____________________________ $m = a_1 \wedge n \geq 1$

i := 2

____________________________ $i = 2 \wedge m = a_1 \wedge n \geq 1$

____________________________ $i \leq n + 1 \wedge m = \max\{a_1, \ldots, a_{i-1}\}$

while i≤n **do**

______________________ $i \leq n \wedge i \leq n + 1 \wedge m = \max\{a_1, \ldots, a_{i-1}\}$

if m<a[i] **then**

_______________ $m < a_i \wedge i \leq n \wedge m = \max\{a_1, \ldots, a_{i-1}\}$

_______________ $\max\{a_1, \ldots, a_{i-1}\} < a_i \wedge i \leq n$

_______________ $\max\{a_1, \ldots, a_i\} = a_i \wedge i \leq n$

m := a[i]

_______________ $m = a_i \wedge \max\{a_1, \ldots, a_i\} = a_i \wedge i \leq n$

_______________ $m = \max\{a_1, \ldots, a_i\} \wedge i \leq n$

endif

_____________________ $(m = \max\{a_1, \ldots, a_i\} \wedge i \leq n) \vee$

$(m > a_i \wedge i \leq n \wedge m = \max\{a_1, \ldots, a_{i-1}\})$

____________________ $(m = \max\{a_1, \ldots, a_i\} \wedge i \le n) \vee$
$(m = \max\{a_1, \ldots, a_i\} \wedge i \le n)$

____________________ $m = \max\{a_1, \ldots, a_i\} \wedge i \le n$

```
i := i+1
```

____________________ $i = i' + 1 \wedge m = \max\{a_1, \ldots, a_{i'}\} \wedge i' \le n$

____________________ $m = \max\{a_1, \ldots, a_{i-1}\} \wedge i \le n + 1$

endwhile

____________________________ $i > n \wedge i \le n + 1 \wedge m = \max\{a_1, \ldots, a_{i-1}\}$

____________________________ $i = n + 1 \wedge m = \max\{a_1, \ldots, a_{i-1}\}$

____________________________ $m = \max\{a_1, \ldots, a_n\}$

10. a. [10] Prove the following code is partially correct with respect to precondition "$m \ge 1$ and $n \ge 1$" and postcondition "$b = \binom{m}{n}$" (assume b, m, n, and k are integer variables.):

```
b :=m
k := 2
while k≤n do
        b := (b*(m-k+1)/k
        k := k+1
endwhile
```

You may use the following axiom:

For $m \ge 1$ and $k \ge 2$, $\binom{m}{k-1} \cdot \frac{m-k+1}{k} = \binom{m}{k}$.

____________________________ $m \ge 1 \wedge n \ge 1$

```
b :=m
```

____________________________ $b = m \wedge m \ge 1 \wedge n \ge 1$

```
k := 2
```

____________________________ $k = 2 \wedge b = m \wedge m \ge 1 \wedge n \ge 1$

____________________________ $2 \le k \le n + 1 \wedge b = \binom{m}{k-1} \wedge m \ge 1$

while k≤n **do**

____________________ $2 \le k \le n \wedge k \le n + 1 \wedge b = \binom{m}{k-1} \wedge m \ge 1$

____________________ $2 \le k \le n \wedge b = \binom{m}{k-1} \wedge m \ge 1$

```
b := (b*(m-k+1)/k
```

____________________ $b = \frac{b' \cdot (m-k+1)}{k} \wedge 2 \le k \le n \wedge b' = \binom{m}{k-1} \wedge m \ge 1$

_____________________ $2 \le k \le n \wedge b = \binom{m}{k} \wedge m \ge 1$

k := k+1

_____________________ $k = k' + 1 \wedge 2 \le k' \le n \wedge b = \binom{m}{k'} \wedge m \ge 1$

_____________________ $2 \le k \le n + 1 \wedge b = \binom{m}{k-1} \wedge m \ge 1$

endwhile

___________________________ $k > n \wedge 2 \le k \le n + 1 \wedge b = \binom{m}{k-1} \wedge m \ge 1$

___________________________ $k = n + 1 \wedge b = \binom{m}{k-1}$

___________________________ $b = \binom{m}{n}$

…b. [5] Prove that the loop terminates.

while k≤n **do**

_____________________T

b := (b*(m-k+1)/k

_____________________T

k := k+1

_____________________ $k = k' + 1$

_____________________ $n - k < n - k'$

endwhile

Since the integer quantity $n - k$ strickly decreases in the loop, it must eventually satisfy $n - k < 0$ which is equivalent to $k > n$, and the loop terminates.

11. [10] Determine the weakest precondition with respect to the postcondition "*z*=10" for the following code (assume *z*, *y*, and *x* are integer variables and that y is defined):

```
x := 3
z := x-y
if y<0 then
        z := 0
else
        z := 2*z
endif
```

wp (**if** *y<0* **then** z := 0 **else**z := 2*z **endif**, $z = 10$)

$= (y < 0 \Rightarrow wp\,(\text{z := 0},\ z = 10)) \wedge (y \ge 0 \Rightarrow wp\,(\text{z := 2*z0},\ z = 10))$

$= (y \ge 0 \vee \textbf{false}) \wedge (y < 0 \vee z = 5)$

$= (y \ge 0) \wedge (y < 0 \vee z = 5)$

$= (y \ge 0 \wedge y < 0) \vee (y \ge 0 \wedge z = 5)$

$=\textbf{false} \vee (y \geq 0 \wedge z = 5)$

$= y \geq 0 \wedge z = 5$

wp (z :=x-y, $y \geq 0 \wedge z = 5$)=($y \geq 0 \wedge x - y = 5$)

wp (x :=3, $y \geq 0 \wedge x - y = 5$)=($y \geq 0 \wedge 3 - y = 5$)

$=(y \geq 0 \wedge -2 = y)$

$=\textbf{false}$

12. [10] Determine the weakest precondition with respect to the postcondition "$y = ax^2 + bx + c$" for the following (assume a, b, c, y, and *x* are integer variables and that a, b, c, and *x* are defined):

```
y := ax
y:= (y+b)*x
y := y+c
```

(Hint: Be careful regarding x =0.)

wp (y := y+c, $y = ax^2 + bx + c$)= $(y + c = ax^2 + bx + c)$

$= (y = ax^2 + bx)$

wp (y := (y+b)*x, $y = ax^2 + bx$)= $((y + b) * x = ax^2 + bx)$

$= (yx + bx = ax^2 + bx)$

$= (yx = ax^2)$

wp (y := a*x, $yx = ax^2$)= $(ax^2 = ax^2)$

= **true**