ICOM 4075 Assignment # 7 SOLUTIONS

(5.1.6) Prove $\forall_{n \geq 1} \sum_{i=1}^{n} (i \cdot i!) = (n+1)! - 1$

Proof by Induction.

<u>Basis</u> $n = 1$

$$\sum_{i=1}^{1} (i \cdot i!) = (1)\cdot(1)! = \underline{\underline{1}}$$

True.

$$(n+1)! - 1 = (1+1)! - 1 = 2! - 1 = 2 - 1 = \underline{\underline{1}}$$

<u>Inductive</u>: $P(k)$: $\sum_{i=1}^{k} (i \cdot i!) = (k+1)! - 1$

Must show $\forall_{k \geq 1} P(k) \rightarrow P(k+1)$

let m an arbitrary integer ≥ 1.

Assume $P(m)$, that is $\sum_{i=1}^{m} (i \cdot i!) = (m+1)! - 1$

$$\sum_{i=1}^{m+1} (i \cdot i!) = (m+1)\cdot(m+1)! + \sum_{i=1}^{m} (i \cdot i!)$$

By Ind. Hyp.

$$= (m+1)(m+1)! + (m+1)! - 1$$

$$= (m+1)! \left((m+1) + 1\right) - 1$$

$$= (m+1)! (m+2) - 1$$

$$= (m+2)! - 1$$

Q.E.D.

5.1.8 Prove $\forall_{n\geq 0} \sum_{i=0}^{n} (2\cdot(-7)^i) = \frac{1-(-7)^{n+1}}{4}$ —

Prove by Induction

Basis $n=0$

$$\sum_{i=0}^{0} (2\cdot(-7)^i) = 2\cdot(-7)^0 = 2\cdot 1 = \underline{\underline{2}}$$

$$\frac{1-(-7)^{0+1}}{4} = \frac{1-(-7)^1}{4} = \frac{1+7}{4} = \frac{8}{4} = \underline{\underline{2}}$$

True.

Inductive: $P(k)$: $\sum_{i=0}^{k} (2\cdot(-7)^i) = \frac{1-(-7)^{k+1}}{4}$

Must prove $\forall_{k\geq 0} (P(k) \to P(k+1))$

Let m an arbitrary integer ≥ 0.

Assume $P(m)$, then prove $P(m+1)$

$$\sum_{i=0}^{m+1} (2\cdot(-7)^i) = \left[2\cdot(-7)^{m+1}\right] + \sum_{i=0}^{m} (2\cdot(-7)^i)$$

By Ind. Hyp.

$$= 2\cdot(-7)^{m+1} + \frac{1-(-7)^{m+1}}{4}$$

$$= \frac{4\cdot 2\cdot(-7)^{m+1} + 1 - (-7)^{m+1}}{4}$$

$$= \frac{(-7)^{m+1}\cdot(8-1) + 1}{4} = \frac{7\cdot(-7)^{m+1} + 1}{4}$$

$$= \frac{-(-7)(-7)^{m+1} + 1}{4} = \frac{1-(-7)^{m+2}}{4}$$

Q.E.D.

5.1.12 Prove $\forall_{n \geq 0} \sum_{i=0}^{n} \left(-\frac{1}{2}\right)^{i} = \frac{2^{n+1} + (-1)^n}{3 \cdot 2^n}$

<u>Basis</u> $n = 0$

$$\sum_{i=0}^{0} \left(-\tfrac{1}{2}\right)^{i} = \left(-\tfrac{1}{2}\right)^{0} = 1$$

True.

$$\frac{2^{n+1} + (-1)^n}{3 \cdot 2^n} = \frac{2^{0+1} + (-1)^0}{3 \cdot 2^0} = \frac{2^1 + 1}{3 \cdot 1} = \frac{3}{3} = 1$$

<u>Inductive</u>: Assume $\left[\sum_{i=0}^{k} \left(-\frac{1}{2}\right)^{i} = \frac{2^{k+1} + (-1)^k}{3 \cdot 2^k}\right]$ for arb. $k \geq 0$. $\leftarrow P(k)$

Prove $P(k+1)$

$$\sum_{i=0}^{k+1} \left(-\tfrac{1}{2}\right)^{i} = \left(-\tfrac{1}{2}\right)^{k+1} + \sum_{i=0}^{k} \left(-\tfrac{1}{2}\right)^{i} \xrightarrow{\text{Ind. Hyp.}} = \left(-\tfrac{1}{2}\right)^{k+1} + \frac{2^{k+1} + (-1)^k}{3 \cdot 2^k}$$

$$= \frac{(-1)^{k+1} \cdot 3 + 2 \cdot 2^{k+1} + 2(-1)^k}{3 \cdot 2^{k+1}} = \frac{2^{k+2} + 3(-1)^{k+1} + 2(-1)(-1)^{k+1}}{3 \cdot 2^{k+1}}$$

$$= \frac{2^{k+2} + 3(-1)^{k+1} - 2(-1)^{k+1}}{3 \cdot 2^{k+1}} = \frac{2^{k+2} + (3-2)(-1)^{k+1}}{3 \cdot 2^{k+1}}$$

$$= \frac{2^{k+2} + (-1)^{k+1}}{3 \cdot 2^{k+1}}$$

Q.E.D.

5.1.32 Prove 3 divides n^3+2n when n is positive integer.

Basis $n=1$

$n^3+2n = (1)^3+2(1) = 1+2 = 3$ $3|3$ so Basis is true.

Inductive: $\forall_{k>0}\, [3 | k^3+2k] \rightarrow [3 | (k+1)^3+2(k+1)]$

Consider an arbitrary integer $m>0$.

Assume $3 | m^3+2m$.

$$(m+1)^3+2(m+1) = m^3+3m^2+3m+1+2m+2 = \underbrace{(m^3+2m)}_{\substack{\text{div by 3}\\ \text{by Ind. Hyp.}}} + \underbrace{(3m^2+3m+3)}_{\text{div by 3.}}$$

This $(m+1)^3+(2m+1)$ is divisible by 3

Q.E.D.

5.1.40 Prove $\left(\bigcap_{i=1}^{n} A_i\right) \cup B = \bigcap_{i=1}^{n} (A_i \cup B) \leftarrow P(n)$

Basis $n=2$

$$\left(\bigcap_{i=1}^{2} A_i\right) \cup B = (A_1 \cap A_2) \cup B \xrightarrow{\text{Distributive Law of Sets}} (A_1 \cup B) \cap (A_2 \cup B) = \bigcap_{i=1}^{2} (A_i \cup B)$$

Inductive Show $\forall_{k\geq 2}\, P(k) \rightarrow P(k+1)$

Let m an arbitrary integer ≥ 2.

Assume $\left(\bigcap_{i=1}^{m} A_i\right) \cup B = \bigcap_{i=1}^{m} (A_i \cup B)$

$$\left(\bigcap_{i=1}^{m+1} A_i\right) \cup B = \left(A_{m+1} \cap \left(\bigcap_{i=1}^{m} A_i\right)\right) \cup B = (A_{m+1} \cup B) \cap \underbrace{\left(\left(\bigcap_{i=1}^{m} A_i\right) \cup B\right)}_{\text{Ind. Hyp.}}$$

$$= (A_{m+1} \cup B) \cap \left(\bigcap_{i=1}^{m} (A_i \cup B)\right) = \bigcap_{i=1}^{m+1} (A_i \cup B)$$

Q.E.D.

5.1. 50 Basis case is false

Let $n=1$

$$\sum_{i=1}^{n} i = \sum_{i=1}^{1} i = 1$$

However

$$\frac{(n+\frac{1}{2})^2}{2} = \frac{(1+\frac{1}{2})^2}{2} = \frac{(\frac{3}{2})^2}{2} = \frac{\frac{9}{4}}{2} = \frac{9}{8} \neq 1.$$

(5.2.4) There are two ways to add 1¢ to any given amount:

Approach #1 • Replace 3 stamps of 3¢ with one stamp of 10¢

Approach #2 • Replace 2 stamps of 10¢ with 7 stamps of 3¢.

In order to guarantee that adding 1¢ to a given amount yields a configurations to which we can continue to add 1¢ increments, the configuration must have at least 6 3¢ stamps. Thus we can try to prove $P(n)$ for $n \geq 18$. using strong induction.

$P(n)$: we can form postage of n cents using 3¢ and 10¢ stamps. ⇐

Basis

Since in the worst case we may have to replace 2 10¢ stamps we must show enough basis cases to be able to assume $P(k-20)$ true. Thus we show $P(k)$ for $18 \leq k \leq 38$.

K	3¢	10¢	K	3¢	10¢
18	6	0	29	3	2
19	3	1	30	0	3
20	0	2	31	7	1
21	7	0	32	4	2
22	4	1	33	1	3
23	1	2	34	8	1
24	8	0	35	5	2
25	5	1	36	2	3
26	2	2	37	9	1
27	9	0	38	6	2
28	6	1			

← 2 choices

Inductive (Strong)

Must show

$$\forall_{k \geq 38} \left\{ \left[\bigwedge_{j=18}^{k} P(j) \right] \rightarrow P(k+1) \right\}$$

and (conjunction)

Let m an arbitrary integer ≥ 38

Assume $P(j)$ true for all $18 \leq j \leq m$.

To form postage of amount $m+1$ from postage of amount m:

Case 1 m's postage includes 3 or more 3¢ stamps.

- Remove 3 3¢ stamps $\rightarrow \geq 29$
- By Ind. Hyp. $P(m-9)$ true
- Add 1 10¢ stamp
- new amount is $m+1$

Case 2 m's postage includes 2 10¢ stamps.

- Remove 2 10¢ stamps
- By Ind. Hyp. $P(m-20)$ = true since $m-20 \geq 18$
- Add 7 3¢ stamps.
- new amount is $m-20+21=m+1$

Q.E.D.

5.2.12 Prove that any positive integer can be written as a sum of distinct powers of 2. Use strong induction

$P(n)$: n can be written as sum of distinct powers of 2.

Basis $P(1)$

$1 = 2^0 \Leftarrow P(1)$ is true.

Inductive Show $\forall_{k>0} [P(1) \wedge P(2) \wedge \ldots \wedge P(k)] \rightarrow P(k+1)$

Take an arbitrary $m > 0$.

Assume $P(1) \wedge P(2) \wedge \ldots \wedge P(m)$

Consider integer $m+1$

Case 1: $m+1$ even

Then $0 < \frac{m+1}{2} < m+1$ is an integer.

By inductive hypothesis $\frac{m+1}{2}$ can be written as sum of distinct powers of 2.

Let's say $\frac{m+1}{2} = \sum_{i=1}^{l} 2^{p_i}$.

Then $m+1 = \sum_{i=1}^{l} 2^{p_i+1}$ so can also be written as sum of powers of 2.

Case 2: $m+1$ odd and $m+1 > 1$

Then m is even

$\frac{m}{2} > 0$ and by ind. hyp. $\frac{m}{2} = \sum_{i=1}^{n} 2^{p_i}$

Then $m = \sum_{i=1}^{n} (2 \cdot 2^{p_i}) = \sum_{i=1}^{n} 2^{p_i+1}$ where $0 < p_i+1$ for $1 \le i \le n$
i.e. 2^0 is not one of the powers.

$m+1 = 2^0 + \sum_{i=1}^{n} 2^{p_i+1}$ and can also be written as sum of distinct powers of 2

Q.E.D.

5.2.34. $P(n,k)$: $$\sum_{j=1}^{n}\left(\prod_{i=0}^{k-1}(j+i)\right) = \frac{\prod_{i=0}^{k}(n+i)}{k+1} \qquad n,k \text{ positive}$$

Proof by Induction on n, k fixed yet arbitrary

<u>Basis</u> $n=1$

left side | $$\sum_{j=1}^{1}\left(\prod_{i=0}^{k-1}(j+i)\right) = \prod_{i=0}^{k-1}(1+i) = k!$$

right side | $$\frac{\prod_{i=0}^{k}(1+i)}{k+1} = \frac{(k+1)!}{k+1} = k!$$

Basis $\equiv$ True.

<u>Inductive</u> Let ℓ be an arbitrary integer ≥ 1, k an arbitrary integer.

Assume $$\sum_{j=1}^{\ell}\left(\prod_{i=0}^{k-1}(j+i)\right) = \prod_{i=0}^{k}(\ell+i),$$ Show $$\sum_{j=1}^{\ell+1}\left(\prod_{i=0}^{k-1}(j+i)\right) = \frac{\prod_{i=0}^{k}(\ell+1+i)}{k+1}$$

$$\sum_{j=1}^{\ell+1}\left(\prod_{i=0}^{k-1}(j+i)\right) = \prod_{i=0}^{k-1}(\ell+1+i) + \sum_{j=1}^{\ell}\left(\prod_{i=0}^{k-1}(j+i)\right)$$

$$= \prod_{i=0}^{k-1}(\ell+1+i) + \frac{\prod_{i=0}^{k}(\ell+i)}{k+1}$$

$$= \frac{(k+1)\prod_{i=1}^{k}(\ell+i) + \prod_{i=0}^{k}(\ell+i)}{k+1} = \frac{(k+1)\prod_{i=1}^{k}(\ell+i) + \ell\prod_{i=1}^{k}(\ell+i)}{k+1}$$

$$= \frac{((k+1)+\ell)\prod_{i=1}^{k}(\ell+i)}{k+1} = \frac{\prod_{i=1}^{k+1}(\ell+i)}{k+1} = \frac{\prod_{i=0}^{k}((\ell+1)+i)}{k+1}$$

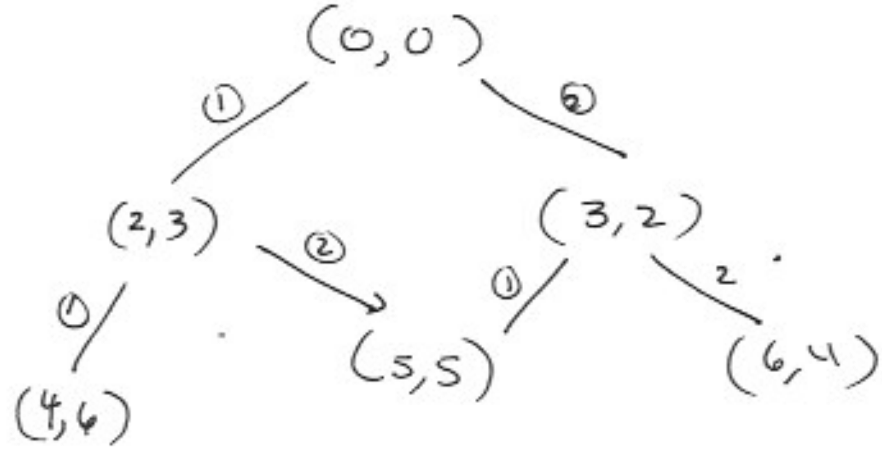
Q.E.D.

5.3.26. Let S be a set of pairs of integers defined recursively as follows:

<u>Basis</u> $(0,0) \in S$

<u>Recursive</u> If $(a,b) \in S$ then ① $(a+2, b+3) \in S$
and ② $(a+3, b+2) \in S$

(a) We will show applications of the recursive rule using a tree



(b) Show $5 \mid a+b$ using strong induction on the number of recursive applications n.

<u>Basis</u> $n=0$ (no recursive applications to $(0,0)$)
$(0,0) \in S$ so $P(0)$ is true.

<u>Inductive</u> Assume that for every (a,b) obtained by k or less recursive applications $5 \mid a+b$.

Must show that any (a,b) obtained by $k+1$ applications also $5 \mid (a+b)$.

Consider an arbitrary pair (a', b') obtained from $k+1$ applications.

<u>Case 1</u>: $(a', b') = (a+3, b+2)$ for some (a,b) from k or less applications.
Since, by Ind. Hyp, $5 \mid (a+b)$, $a+b = 5k$. $a'+b' = a+3+b+2 = (a+b)+5 = 5k+5$.
Thus $5 \mid a'+b'$.

<u>Case 2</u>: Almost identical.

Q.E.D.

5.3.26.c Prove $5 \mid a+b$ when $(a,b) \in S$

<u>Basis</u> $(0,0) \in S$

$0+0=0$ and $5 \mid 0$ so basis is true.

<u>Inductive</u> Consider an arbitrary $(a,b) \in S$.

Assume $5 \mid a+b$.

<u>Case 1</u>: Let $(a',b') = (a+3, b+2)$

$$a'+b' = a+3+b+2 = \underbrace{5}_{\substack{5 \\ \text{divides} \\ 5}} + \underbrace{(a+b)}_{\substack{5 \text{ divides} \\ \text{by} \\ \text{Ind Hyp.}}}$$

<u>Case 2</u>: Let $(a',b') = (a+2, b+3)$

$$a'+b' = a+2+b+3 = \underbrace{(a+b)}_{\substack{\text{div by} \\ 5 \\ \text{by Ind Hyp.}}} + \underbrace{5}_{\substack{\text{div} \\ \text{by} \\ 5.}}$$

Q. E. D.

5.3.32. Recall the recursive definition of Σ^*

Basis $\lambda \in \Sigma^*$

Recursive if $w \in \Sigma^*$ and $x \in \Sigma$ then $wx \in \Sigma^*$

(a) Recursive definition of $ones(w)$ where $w \in \Sigma^*$

Basis $ones(\lambda) = 0$

Recursive if $w \in \Sigma^*$, $x \in \Sigma$.

$$ones(w1) = ones(w) + 1$$
$$ones(w\emptyset) = ones(w)$$

(c) Prove $ones(st) = ones(s) + ones(t)$ $\forall s,t \in \Sigma^*$.

Basis let $s = t = \lambda$

$ones(st) = ones(\lambda) = \underline{0}$

$ones(s) + ones(t) = 0 + 0 = \underline{0}$

Inductive let $s,t \in \Sigma^*$ $s,t \neq \lambda$. ~~let x~~

We will consider four cases: $ones(s1t)$, $ones(s\emptyset t)$, $ones(st1)$, $ones(st\emptyset)$.

Assume $ones(st) = ones(s) + ones(t)$

Case 1: $s\emptyset t$ or $st\emptyset$

$$ones(s\emptyset t) = ones(st\emptyset) = ones(st) + \emptyset = \overbrace{ones(s) + ones(t)}^{\text{Ind. Hyp.}}$$

$$ones(s\emptyset) + ones(t) = ones(s) + ones(t\emptyset) = ones(s) + ones(t)$$

Case 2: $s1t$ or $st1$.

$$ones(s1t) = ones(st1) = ones(st) + 1 = ones(s) + ones(t) + 1$$

$$ones(s1) + ones(t) = ones(s) + ones(t1) = ones(s) + ones(t) + 1$$

Q.E.D.

(5.4.12) procedure powerm (x, n, m : positive integers)

```
if n=0
    return 1
else
    return ((x mod m) . (powerm (x, n-1, m))) mod m
```

(5.4.16) procedure sumfirstN (n : integer)

```
if n=1
    return 1
else
    return (n + sumfirstN (n-1))
```

Proof of Correctness:

Basis sumFirstN (1) = 1

Inductive Assume sumfirstN (k) returns the sum of the first k integers.

sumFirstN (k+1) will return (k+1) + sumFirstN (k)

By Ind. Hyp. sumFirstN (k) is the sum of the first k ints.

Therefore (k+1) + sumFirstN (k) is the sum of the first k+1 ints.

(5.4.34) We develop a naive recursive algorithm straight from the recurrence relation:

```
procedure sequence (n)
  if n=0
     return 1
  else if n=1
     return 2
  else if n=2
     return 3
  else return sequence (n-1)+sequence(n-2)+sequence(n-3).
```

As show for the recursive fibonacci algorithm, the algorithm will yield exponential growth.
However ~~but~~ this doesn't mean that all recursive algorithms are this bad. We could think of a recursive algorithm that is just as efficient as an efficient iterative algorithm.

```
procedure sequence (n, f0, f1, f2)
   if n=0 return f0
   if n=1 return f1
   if n=2 return f2
   else return (n-1, f1, f2, f0+f1+f2)
```

This algorithm is $\Theta(n)$, just like the best iterative algorithm.

5.4.38 procedure powerString (w: string; i: integer)

```
if i=0
    return λ
else
    return w + powerString(w, i-1)
```

5.4.40. Prove algorithm in 5.4.38 is correct.

Basis powerString(w, ϕ)

returns λ which correspond with zero copies of w.

Inductive Assume powerString (w, k) works. for some $k \geq 0$

powerString(w, k) returns k copies of w.

powerString(w, $k+1$) returns w + powerString (w, $k+1-1$)

And this is w + powerString(w, k), Therefore returns $k+1$ copies of w.

5.4.52

```
procedure quickSort (a1, a2, ..., an, i, j: integers)
  if i < j
    pivot = ai
    partition (a1, a2 ... an, i, j, ~~pivot~~)
    ~~quick~~ let k position of ai in partitioned array.
    quickSort (a1, a2, ..., an, i, k-1)
    quickSort (a1, a2, ..., an, k+1, j)
```

5.4.52

```
procedure quickSort (a1, ... an, i, j : integer)
if (i<j)
    pivot = partition (a1, ..., an, i, j)
    quickSort (a1, a2, ..., an, i, pivot-1)
    quickSort (a1, a2, ..., an, pivot+1, j)

procedure partition (a1, a2, ..., an, i, j)
    // en.wikipedia.org/wiki/QuickSort.
    pivotIndex = i
    pivotValue = a_i
    swap (a_i, a_j)
    storeIndex = i
    for l = i to j-1
        if a_l < pivotValue
            swap (a_l, a_storeIndex)
            storeIndex = storeIndex + 1
    swap (a_storeIndex, a_j)
    return storeIndex.
```