HW 5

6.5.1 (b) Prove that for any positive integer n, 6 evenly divides $7^n - 1$.

Proof.

By induction on n.

Base case:

$n = 1$, $7^n - 1 = 6$, 6 evenly divides 6.

Inductive step:

Assum that k is a positive integer and 6 evenly divides $7^n - 1$, and prove that 6 evenly divides $7^n - 1$. Since 6 evenly divides $7^n - 1$, $7^n - 1 = 6m$ for some integer m, and therefore $7^n - 1 = 6m$.

$$7^{(k+1)} - 1 = 7^{k+1} - 1$$
$$= (7^k \times 7^1) - 1$$
$$= (7^k \times 7) - 1 - 6 + 6$$
$$= ((7^k \times 7) - 7) + 6$$
$$= 7(7^k - 1) - 6$$
$$= 7 \times 6m - 6 \quad \text{by the inductive hypothesis}$$
$$= 6(7m - 1)$$

Therefore, $7^{k+1} - 1$ is a multiple of 6, and 6 evenly divides $7^{k+1} - 1$. ■

6.7

6.6.1 (b) Prove that any amount of postage worth 24 cents or more can be made from 7-cent or 5-cent stamps.

Proof.

By induction on the amount of postage.

Base case:

- 24 cents: use two 7-cent and two 5-cent stamp.
- 25 cents: use five 5-cent stamp.
- 26 cents: use a 5-cent and three 7-cent stamp.
- 27 cents: use four 5-cent and a 7 cent stamp.
- 28 cents: use four 7-cent stamp.

Inductive step :

Assume that for $k \geq 28$, it is possible to make j cents worth of postage using only 7-cent and 5-cent stamps for any j in the range form 24 through k, Show that it is possible to make $k+1$ cents worth of postage using only 7-cent and 5-cent stamps.

Since $k \geq 28$, then $k-4 \geq 24$ therefore $k-4$ falls in the range from 24 through k, and by the inductive hypothesis, it is possible to make $k-4$ cents worth of stamps using only 7-cent and 5-cent stamps. By adding one 5-cent stamp, the amount of postage becomes $(k-4)+5 = k+1$. There fore, it is possible to make $k+1$ cents worth of postage using only 7-cent and 5-cent stamps.

Also, in a function of k to show

$k = 5a + 7b$

$k+1 = 5a+7b+1$

$= 5a+7b+21-20$

$= 5(a-4)+7(b+3)$.

There fore, $5(a-4)+7(b+3)$ is another type of $5a+7b$, so k is approved by the expansion of $k+1$. can be made 5-cent and 7-cent stamps. ∎

1.7.1 (b) The loop below computes the sum of a list of numbers.

```
while (j<n)
    sum := sum + a_{j+1}
    j := j+1
End-while
```

- Pre-condition : $j=1$, $sum = a_1$, n is a positive integer, $a_1, \dots, a_n$ is a list of n numbers.
- Post-condition : sum $\sum_{k=1}^{n} a_k$
- loop variant : j is an integer, $j \leq n$, and sum $= \sum_{k=1}^{j} a_k$

Step 1. Assume that $j=1$, $sum = a_1$, n is a positive integer, We will prove that j is an integer, $j \leq n$, and sum $=$ $\sum_{k=1}^{j} a_k$

Since n is a positive integer, $0 < n$. Since $j=1$, j is a integer. Since $j=1$, then $j \leq n$. Finally since $j=1$ and $sum = a_1$, $sum = \sum_{k=1}^{j} a_k$ for any k.

Step 2. Let j_1 and sum_1 be the values of j and sum before an iteration of the loop. Let j_2 and sum_2 be the values of j and sum after the iterations of the loop. Assume that $j_1 < n$, j_1 is an integer, and $sum_1 = \sum_{k=1}^{j_1} a_k$. We will prove that j_2 is an integer, $j_2 \le n$, and $sum_2 = \sum_{k=1}^{j_2} a_k$.

Since j is incremented by 1 in the loop, $j_2 = j_1 + a_j + 1$, and since sum is incremented by k in the loop, $sum_2 = sum_1 + a_{k+1}$

Since j_1 is an integer and $j_2 = j_1 + 1$, then j_2 is also an integer. Also, since j_1 is an integer and $j_1 < n$, then $j_1 \le n-1$. Adding 1 to both sides of the inequality gives $j_1 + 1 \le n$, which is equivalent to $j_2 \le n$. Finally, $sum_2 = sum_1 + a_{k+1} = \sum_{k=1}^{j_1} a_k + a_{k+1}$. Let a_{k+1} be the last value in the summation. So the summation is true. As the while loop showed before, sum $=$ sum $+ a_{j+1}$, plug into formula, sum $= \sum_{k=1}^{j_1} a_k$, $\therefore a_{k+1} = a_{j+1}$, and $j_1 = j+1$, $\sum_{k=1}^{j_1} a_k + a_{k+1} = \sum_{k=1}^{j_1} a_k + a_{j+1} = \sum_{k=1}^{j_1+1} a_k$, $j_1 + 1 = j_2$, $\therefore sum_2 = \sum_{k=1}^{j_2} a_k$.

Step 3. After a finite number of iterations $j \ge n$. The loop eventually terminates because after n iterations, the value of j is n, which means that the condition $j < n$ will be false.

Step 4. Assume that the condition $j < n$ is false (which means that $j \ge n$), $j \le n$, and sum $= \sum_{k=1}^{j} a_k$. We will prove that sum $= \sum_{k=1}^{n} a_k$.

Since $j \le n$ and $j \ge n$, then $j = n$. Therefore sum $= \sum_{k=1}^{j} a_k = \sum_{k=1}^{n} a_k$.

$26 + 4 + 10 = 40$

7.1.2 (a) Total 40 characters in these list. 26 letters 10 digits, 4 special characters

length 6 : $\boxed{40^6} = 40 \times 40 \times 40 \times 40 \times 40 \times 40$

(b) length 7 : $\boxed{40^7} = 40 \times 40 \times 40 \times 40 \times 40 \times 40 \times 40$

length 8 : $\boxed{40^8} = 40 \times 40 \times 40 \times 40 \times 40 \times 40 \times 40 \times 40$

length 9 : $\boxed{40^9} = 40 \times 40 \times 40 \times 40 \times 40 \times 40 \times 40 \times 40 \times 40$

(c) length 7 : $(40-26) \times 40^{7-1} = \boxed{14 \times 40^6}$

length 8 : $(40-26) \times 40^{8-1} = \boxed{14 \times 40^7}$

length 9 : $(40-26) \times 40^{9-1} = \boxed{14 \times 40^8}$

7.1.4 (a) 4-printers, 100 pages $= \boxed{4^{100}}$

(b) 2 color printers, 2 pages (first and last page) in color

$2 \times 2 = 4$

100 pages − 2 colored pages = 98 pages,

2 color printer can also print black-and-white

$\boxed{4^{98} \times 4}$

(c) $100 \div 25 = 4$ stacks of paper, each printer can print either one of four stacks,

4 printers

$\boxed{4^4}$

7.2.2 (c) Define the function $f: P_7 \to B^4$ such that if $x \in P_7$, the $f(x)$ is obtained by dropping the last three bits of x. For example $f(0111110) = 0111$.
Notice that if $x \in P_7$, then the first and seventh bits of x are the same, the second and sixth bits are the same, and the third and fifth bits of x are the same, so $x = b_1 b_2 b_3 b_4 b_3 b_2 b_1$, where each $b_i = 0$ or 1. f is onto, because for any $y \in B^4$, yy^R is a palindrome and $f(yy^R) = y$. f is one-to-one because for x and $y \in P_7$, if $f(x) = f(y)$, then x and y have the same first three bits. Since the last three bits of a palindrome of length 7 are determined by the first three bits, then x and y must be equal.

Since there is a bijection from P_7 to B^4, $|P_7| = |B^4| = \boxed{2^4}$

7.2.5 (a) $f: S \to T$, $x \in S$, $f(x) = x$ Gene > Don
In the line up, Gene is ahead of Don, $f(x) = x$.
$\therefore$ the output will be the same as input:
f(Fran, Gene, Hal, Jan, Abe, Don, Cam, Eli, Ike, Ben)

(b) In this line up, Don is ahead of Gene, then $f(x)$ is ordering that is same as x, (except that Don and Gene have swapped places)
$\therefore$ the output will be the same as input:
f(Eli, Ike, Don, Hal, Jan, Abe, Ben, Fran, Gene, Cam)

(c) Yes, the function f is a k-to-1 correspondence for some positive integer k, because two line-ups can have the same output.

If considering $f(x)$ as Gene is ahead of Don and
$f'(x)$ as Don is ahead of Gene (since swapped places)
$\because f(x) = x \in S$
$\therefore f'(x) = x' \in S$, $x \neq x'$ ($f(x) \neq f'(x)$)

Therefore, $\boxed{k=2}$, for function f a k-to-1 correspondence, f can be f(x) or f'(x).

(d) $|S| = 3628800$, $k=2$ $\qquad 10! = 3628800$

$\therefore f: S \to T$ is a 2-to-1 correspondence,

$|S| = |T| \times 2$

$|T| = 3628800 \div 2$

$\boxed{|T| = 1814400}$

(e) $f: S \to Q$

Gene befor Don, Jan before Abe

$k=2$ $\qquad k=2$

$k = 2+2 = 4$

$\therefore$ 4-to-1 correspondence

$|S| = |Q| \times 4$

$|Q| = 3628800 \div 4$

$\boxed{|Q| = 907,200}$

7.3.3 (a) Digit = 10 , letters = 26 D for digit , L for letters

$10 \times 26 \times 26 \times 26 \times 26 \times 10 \times 10 = \boxed{10^3 \times 26^4}$

D L L L L D D

(b) $10 \times 26 \times 26 \times 26 \times 26 \times 9 \times 8 = \boxed{10 \times 9 \times 8 \times 26^4}$

D L L L L D D

(c) $\boxed{10 \times 26 \times 25 \times 24 \times 23 \times 9 \times 8}$

D L L L L D D

7.3.4 (a) 3 projects to 3 coders. 7 junior, 3 senior, one person, one project

$7+3 = 10$

possible ways : project 1 = 8 $(10-2=8)$

Project 2 = 3

Project 3 = 7 two coders selected first

$8 \times 3 \times 7 = \boxed{168}$