I. Theorem: $\boldsymbol{A^N = 1}$ **for all non-zero real numbers** $\boldsymbol{A}$**.**

A. Basis Step with $N = 0$: $A^0 = 1$ for all non-zero numbers A.

B. Inductive Assumption: $A^k = 1$ for all natural numbers k.

C. Then: $A^{k+1} = \frac{A^k \times A^k}{A^{k-1}} = \frac{1 \times 1}{1} = 1$

so: $A^N = 1$ for all non-zero real numbers A.

D. Flaws:

1. The proof makes use of A^{k-1} in addition to A^k in the inductive step so it is really a proof using strong induction.

2. $A^1 = A$ for all non-zero real numbers A.
   Therefore $P(0)$ does **not** imply $P(1)$.

E. The basis step for $P(0)$ is not a proper basis step in this case.

II. Theorem: Every postage of three cents or more can be formed using just three-cent and four-cent stamps.

OR

For $N \geq 3$ $N = 3 \times I + 4 \times J$ where I and J are integers.

B. Basis Step: $3 = 3 \times 1 + 4 \times 0 \, 4 = 3 \times 0 + 4 \times 1$

C. Inductive Step:

Assume $N = 3 \times I + 4 \times J$ for $N \geq 3$

Then
$N + 1 = 3 \times I + 4 \times J + 1 = 3 \times (I - 1) + 4 \times (J + 1)$

D. Flaw: $5 \neq 3 \times I + 4 \times J$ for any $I, \ J$
Therefore, the basis step does not imply the next step.

E. Proofs of statements regarding subsets of the natural numbers must have the basis step specified as the first element of the subset of the natural numbers which includes all elements for which the statement is true.

F. **Note:** The theorem:

Every postage of six cents or more can be formed using just three-cent and four-cent stamps.

or:

For $N \geq 6$ $N = 3 \times I + 4 \times J$ where I and J are integers.

is proveably true.

III. **Example:** Determine which amounts of postage can be formed using just 5-cent and 6-cent stamps and prove your answer to using the principle of mathematical induction.

A. Pre-Basis Step:

1. The following amounts can be formed using 5-cent and 6-cent stamps:

0	=	0 * 6	+	0 * 5
5	=	0 * 6	+	1 * 5
6	=	1 * 6	+	0 * 5
10	=	0 * 6	+	2 * 5
11	=	1 * 6	+	1 * 5
12	=	2 * 6	+	0 * 5
15	=	0 * 6	+	3 * 5
16	=	1 * 6	+	2 * 5
17	=	2 * 6	+	1 * 5
18	=	3 * 6	+	0 * 5

2. The amounts of 1, 2, 3, 4, 7, 8, 9, 13, 14, and 19 cannot be formed using 5-cent and 6-cent stamps.

   Therefore: $P(N) \equiv$ All amounts $N \geq 20$ can be formed using 5-cent and 6-cent stamps.

   or: $P(N) \equiv N = 6 \times k + 5 \times j$ where k and j are integers and $N \geq 20$.

   Note: A proof using a basis step of :

   a. $N \geq 5$ is flawed since the theorem is not true for $N = 78,\ 9,\ 13,\ 14,$ and $19.$

   b. $N \geq 10$ is flawed since the theorem is not true for $N = 13,\ 14,$ and $19.$

   c. $N \geq 15$ is flawed since the theorem is not true for $N = 19.$

C Basis step(s):

$N = 20$ $20 = 0 \times 6 + 4 \times 5$ so $k = 0$ and $j = 4$

$N = 21$ Remove one 5-cent stamp and replace it with a 6-cent stamp

$N = 22$ Remove two 5-cent stamps and replace them with two 6-cent stamps.

$N = 23$ Remove three 5-cent stamps and replace them with three 6-cent stamps.

$N = 24$ Remove all 5-cent stamps and replace them with four 6-cent stamps.

$N = 25$ Add one 5-cent stamp to the original collection for 20.

$N = 26$ Remove one 5-cent stamp from the collection for 25 and replace it with one 6-cent stamp.

D. Inductive step: We assume that $P(N)$ is true for $20 \leq N \leq 26$

1. In general, since $N = 6 \times k + 5 \times j$, where k and j are integers, we can pick one of the following alternatives for constructing $N+1$, one of which must work given our formulation of the basis step.

   a. Repeatedly subtract 5 from $N{+}1$, keeping count of the number of subtractions M, until a result that is divisible by 6 is reached. Substitute one of the 5's removed with a 6 and re-add the remaining 5's removed to get:

   $$N+1 = \left[\tfrac{[N-((M-1)\times 5)]}{6} + 1\right] \times 6 + (M-1) \times 5$$

   b. Repeatedly subtract 6 from N, keeping count of the number of subtractions P, until a result that is divisible by 5 is reached. Substitute four of the 6's removed with five 5's and re-add the remaining 6's removed to get:

   Then: $N+1 = (P-4) \times 6 + \left[\tfrac{[N-((P-1)\times 6)]}{5} + 5\right] \times 5$

2. We have, then, shown that: $P(20)$ is true and $P(N) \Rightarrow P(N+1)$

   Hence $P(N) \equiv$ All amounts $N \geq 20$ can be formed using 5-cent and 6-cent stamps

   has been proven using the Principle of Mathematical Induction.

8. The result shown above has been verified by the program code which follows.

E. A computation of the amounts that can be created using five and six cent stamps.

1. Code

```
/*
 * A computation of the amounts that can be created using five
 * and six cent stamps.
 */
package fiveandsix;

import  java.util.Scanner;

public class FiveAndSix
   {
   final   static  long ZERO    =   0;
   final   static  long ONE     =   1;
   final   static  long TWO     =   2;
   final   static  long FIVE    =   5;
   final   static  long SIX     =   6;

   /*
    *  This is the main program from which all others all invoked.  It
    *  sets the initial values and enters the first two values in the
    *  solution set.
   */
   public static void main(String[] args)
      {
      long total      =   ZERO;
      long[] cntArray  =   new long[4];
      Scanner keyBoard = new Scanner(System.in);

      System.out.print("Enter a starting integer for testing:  ");
      long    Start   =   keyBoard.nextLong();
      System.out.print("Enter an integer at which testing ends:  ");
      long    End     =   keyBoard.nextLong();

      if (Start < 20) Start = ZERO;
```

```
    for (long i = Start; i <= End; i++)
       {
       if (fCounts(i, SIX, FIVE, cntArray))
          System.out.printf("%8d %6c %11d  %9s %7d %4s\n",
                cntArray[0], '=', cntArray[1], "* 6        +",
                cntArray[2], " * 5" + "   divisor =  " + cntArray[3]);
       else System.out.println(i + " does not meet the criteria");
      }
    }

/*
 *  This function performs the main task of solution computation.
 *  It checks each value between the initial solution indices and the
 *  limit prescribed by the main program
*/

public static boolean fCounts(long intVal, long Fact1, long Fact2,
                        long[] cntVals)
    {
    cntVals[0] = ZERO;
    cntVals[1] = ZERO;

    long Rem1 = intVal % Fact1;
    long Rem2 = intVal % Fact2;

    if ((Rem1 == ZERO) || (Rem2 == ZERO))
       {
       if (Rem1 == ZERO)   {
                   cntVals[0] = intVal;
                   cntVals[1] = intVal / Fact1;
                   cntVals[2] = ZERO;
                   cntVals[3] = ZERO;
                   return true;
                   }
```

```
        else if (Rem2 == ZERO)  {
                    cntVals[0] = intVal;
                    cntVals[1] = ZERO;
                    cntVals[2] = intVal / Fact2;
                    cntVals[3] = ZERO;
                    return true;
                    }
        else;
        }

      else    {
          boolean solFound    =   false;
          long     Fives   =   ZERO;
          long     Sixes   =   ZERO;

          long     T       =   intVal;

          while ((T >= Fact2) && !solFound)
            {
            T = T - Fact1;
            Sixes++;
            Rem2    =   T % Fact2;
            if (Rem2 == ZERO)   {
                        cntVals[0]  =   intVal;
                        cntVals[1]  =   Sixes;
                        cntVals[2]  =   T / Fact2;
                        cntVals[3]  =   Fact1;
                        solFound    =   true;
                        return  solFound;
                        }
            else;
            }
```

```
            T   =   intVal;

            while ((T >= Fact1) && !solFound)
               {
               T = T - Fact2;
               Fives++;
               Rem1    =   T % Fact1;
               if (Rem1 == ZERO)   {
                              cntVals[0]  =   intVal;
                              cntVals[1]  =   T / Fact1;
                              cntVals[2]  =   Fives;
                              cntVals[3]  =   Fact2;
                              solFound    =   true;
                              return  solFound;
                              }
               else;
               }
         }
      return false;
      }
}
```

2. Output

```
run:
Enter a starting integer for testing:  0
Enter an integer at which testing ends:  75
     0      =          0  * 6       +       0  * 5   divisor =  0
1 does not meet the criteria
2 does not meet the criteria
3 does not meet the criteria
4 does not meet the criteria
     5      =          0  * 6       +       1  * 5   divisor =  0
     6      =          1  * 6       +       0  * 5   divisor =  0
7 does not meet the criteria
8 does not meet the criteria
9 does not meet the criteria
```

10	=	0 * 6	+	2 * 5	divisor = 0
11	=	1 * 6	+	1 * 5	divisor = 6
12	=	2 * 6	+	0 * 5	divisor = 0
13 does not meet the criteria					
14 does not meet the criteria					
15	=	0 * 6	+	3 * 5	divisor = 0
16	=	1 * 6	+	2 * 5	divisor = 6
17	=	2 * 6	+	1 * 5	divisor = 6
18	=	3 * 6	+	0 * 5	divisor = 0
19 does not meet the criteria					
20	=	0 * 6	+	4 * 5	divisor = 0
21	=	1 * 6	+	3 * 5	divisor = 6
22	=	2 * 6	+	2 * 5	divisor = 6
23	=	3 * 6	+	1 * 5	divisor = 6
24	=	4 * 6	+	0 * 5	divisor = 0
25	=	0 * 6	+	5 * 5	divisor = 0
26	=	1 * 6	+	4 * 5	divisor = 6
27	=	2 * 6	+	3 * 5	divisor = 6
28	=	3 * 6	+	2 * 5	divisor = 6
29	=	4 * 6	+	1 * 5	divisor = 6
30	=	5 * 6	+	0 * 5	divisor = 0
31	=	1 * 6	+	5 * 5	divisor = 6
32	=	2 * 6	+	4 * 5	divisor = 6
33	=	3 * 6	+	3 * 5	divisor = 6
34	=	4 * 6	+	2 * 5	divisor = 6
35	=	0 * 6	+	7 * 5	divisor = 0
36	=	6 * 6	+	0 * 5	divisor = 0
37	=	2 * 6	+	5 * 5	divisor = 6
38	=	3 * 6	+	4 * 5	divisor = 6
39	=	4 * 6	+	3 * 5	divisor = 6
40	=	0 * 6	+	8 * 5	divisor = 0
41	=	1 * 6	+	7 * 5	divisor = 6
42	=	7 * 6	+	0 * 5	divisor = 0
43	=	3 * 6	+	5 * 5	divisor = 6
44	=	4 * 6	+	4 * 5	divisor = 6
45	=	0 * 6	+	9 * 5	divisor = 0
46	=	1 * 6	+	8 * 5	divisor = 6

```
47 = 2 * 6 + 7 * 5 divisor = 6
48 = 8 * 6 + 0 * 5 divisor = 0
49 = 4 * 6 + 5 * 5 divisor = 6
50 = 0 * 6 + 10 * 5 divisor = 0
51 = 1 * 6 + 9 * 5 divisor = 6
52 = 2 * 6 + 8 * 5 divisor = 6
53 = 3 * 6 + 7 * 5 divisor = 6
54 = 9 * 6 + 0 * 5 divisor = 0
55 = 0 * 6 + 11 * 5 divisor = 0
56 = 1 * 6 + 10 * 5 divisor = 6
57 = 2 * 6 + 9 * 5 divisor = 6
58 = 3 * 6 + 8 * 5 divisor = 6
59 = 4 * 6 + 7 * 5 divisor = 6
60 = 10 * 6 + 0 * 5 divisor = 0
61 = 1 * 6 + 11 * 5 divisor = 6
62 = 2 * 6 + 10 * 5 divisor = 6
63 = 3 * 6 + 9 * 5 divisor = 6
64 = 4 * 6 + 8 * 5 divisor = 6
65 = 0 * 6 + 13 * 5 divisor = 0
66 = 11 * 6 + 0 * 5 divisor = 0
67 = 2 * 6 + 11 * 5 divisor = 6
68 = 3 * 6 + 10 * 5 divisor = 6
69 = 4 * 6 + 9 * 5 divisor = 6
70 = 0 * 6 + 14 * 5 divisor = 0
71 = 1 * 6 + 13 * 5 divisor = 6
72 = 12 * 6 + 0 * 5 divisor = 0
73 = 3 * 6 + 11 * 5 divisor = 6
74 = 4 * 6 + 10 * 5 divisor = 6
75 = 0 * 6 + 15 * 5 divisor = 0
BUILD SUCCESSFUL (total time: 18 seconds)
```

IV. Example 4: For every positive integer N, $\sum_{i=1}^{N} i = \frac{(N+\frac{1}{2})^2}{2}$

A. Inductive Step:

1. Assume: $\sum_{i=1}^{N} i = \frac{(N+\frac{1}{2})^2}{2}$

2. Then: $\sum_{i=1}^{N+1} i = \sum_{i=1}^{N} i + (N+1) = \frac{(N+\frac{1}{2})^2}{2} + (N+1)$

$$= \frac{N^2+N+\frac{1}{4}+2N+2}{2} = \frac{N^2+3N+\frac{9}{4}}{2}$$

$$\frac{\left[(N+1)+\frac{1}{2}\right]^2}{2} = \frac{(N+\frac{3}{2})^2}{2} = \frac{N^2+3N+\frac{9}{4}}{2}$$

so: $P(N) \rightarrow P(N+1)$ □

B. Flaw:

Basis step does not hold: $\sum_{i=1}^{1} i = \frac{(1+\frac{1}{2})^2}{2} = \frac{\frac{9}{4}}{2} = \frac{9}{8} \neq 1$

C. Since the basis step does not hold, the proof is meaningless.

V. Example 5: All horses are the same color.

A. Basis Step:

1. For $N = 1$ we have one horse which the same color as itself.

2. Basis step is clearly true.

B. Inductive Assumption: For some $N > 1$, assume that a remuda of N horses are all the same color.

C. Inductive Proof: Consider a remuda of $N+1$ horses.

1. By the inductive assumption the horses $H_1,\ H_2,\ H_3, ..., H_N$ are all the same color.

2. But, since our inductive assumption applies to any group of N horses then the horses $H_2,\ H_3, ..., H_N, H_{N+1}$ must also be of the same color.

3. Therefore the remuda of $N+1$ horses are all of the same color.

D. Therefore, $P(1)$ is true and $P(N) \to P(N+1)$ so we have proven that all horses are the same color.

E. Flaws:

1. Basis Step:
   a. The statement "All horses are the same color." implies a plurality of hourses.
   b. Therefore the basis step should involve two horses.
   c. The basis step would, then, fail.

2. Inductive Step:
   a. The inductive step was executed by choosing N horses out of $N+1$ and stating that they were of the same color.

   b. This works only if the inductive assumption is that all $N+1$ horses are of the same color.

   c. Improper inductive logic.

VI. Example 6: For any collection of N lines in the plane, if no two are parallel, then all lines intersect at one point.

A. Basis Step: For $N = 2$ the statement is obviously true since two non-parallel lines can only intersect in one point.

B. Inductive Assumption: For any collection of N lines in the plane, if no two are parallel, then all lines intersect at one point.

C. Inductive Proof:

1. Choose k_1 and k_2 such that:
   a. $k_1 < N$ and $k_2 < N$

   b. $k_1 + k_2 = N + 1$

2. By our inductive assumption all of the lines in the collection of k_1 lines intersect at one point.

3. By our inductive assumption all of the lines in the collection of k_2 lines intersect at one point.

4. Therefore we have a collection of $N + 1$ lines that intersect at one point.

D. Flaws:

1. No evidence that the $k_1$lines intersect at the same point as the k_2 lines.

2. The statement that the $N + 1$ lines intersect at the same point is the same as making the inductive assumption for $N + 1$ lines.

VII. Example 7: All positive integers are equal.

A. Basis Step: $N = 1$

1. x, y, $N \in Z^+$ and $max(x, y) = N$

2. Therefore: $x = y = 1$

B Inductive Assumption: $max(x, y) = N \rightarrow x = y$

C. Inductive Step:

1. Consider two positive integers $x,\ y$, $N \in Z^+$ such that $max(x, y) = N + 1$

2. Then: $max(x - 1, y - 1) = N$

3. By our inductive assumption $x - 1 = y - 1$

4. This requires that: $x = y$

D. Therefore all positive integers are equal. □

E. Flaws:

1. For $N \geq 2$ the theorem does not hold since:
   a. $x = 1$ and $y \geq 2$

   b. $max(x, y) = N$ does not imply that $x = y$

2. The inductive step considers only the subset of Z^+ for which $|x - y| = 1$.
   a. It needs to consider $x,\ y$ for which $|x - y| \geq 1$

   b. For these cases $max(x, y) = N + 1$ does not imply that $x = y$.