# Day 7 - Programs at Bootcamp

**Section A - Elements of Programing :- Condition, Loops and Logical Programming**

1. Write a program ***Binary.java*** prints the binary (base 2) representation of the decimal number typed as the command-line argument. It is based on decomposing the number into a sum of powers of 2. For example, the binary representation of 106 is 11010102, which is the same as saying that 106 = 64 + 32 + 8 + 2. Ensure necessary padding to represent 4 Byte String.

   To compute the binary representation of n, we consider the powers of 2 less than or equal to n in decreasing order to determine which belong in the binary decomposition (and therefore correspond to a 1 bit in the binary representation).

2. Extend Binary.java to read an integer as an Input, convert to Binary and perform the following functions.

   i. Swap nibbles and find the new number.

   ii. Find the resultant number is the number is a power of 2.

   A nibble is a four-bit aggregation, or half an octet. There are two nibbles in a byte. Given a byte, swap the two nibbles in it. For example 100 is to be represented as 01100100 in a byte (or 8 bits). The two nibbles are (0110) and (0100). If we swap the two nibbles, we get 01000110 which is 70 in decimal.

# Day 7 - Programs at Home

**Section A - Elements of Programing :- Condition, Loops and Logical Programming**

1. Write a program to find the 2nd largest and the 2nd smallest element from an unsorted array and without sorting the array.
2. Write a program to compute Factors of a number N using prime factorization method.

   Logic -> Traverse till i*i <= N instead of i <= N for efficiency.

   O/P -> Print the prime factors of number N.