Aum Amriteswaryai Namah

Programming patterns that we have come across so far.

1. Divisibility

A common scenario that arises in problem solving is checking the divisibility of a number. The % (or mod) operator is a indispensible for this.

Special cases of divisibility includes:
- Even/Odd checking - divisibility by 2
- Extracting digit(s) - divisibility by 10

Example problems: Frog jumping, Find Divisible, Dice Rolling, Uniform String, Petya and Origami, Coins, Sum of digits

2. Generate and Filter

An often useful piece of code is to generate numbers within a range and filter them based on some condition.
- Single loop is used to generate a series of numbers for a given l and r.
- Nested loop is used to generate a series of number pairs for a given l and r.

Example problems: Ehab and Another Construction, Find Divisible, Uniform String

3. Recognizing the end of input

Sometimes you have to read a sequence of chars until a newline or whitespace or EOF is reached.

Example problems: Romaji

4. Skipping chars while reading

At times, one may have to read a sequence of chars, processing a few and skipping the rest based on some criteria.

Example problems: Repeating cipher, Gennady and Card Game.

5. Accumulator

Many a times you have to compute a result by repeated computations over an iteration. Each iteration will take you closer to the result. An accumulator variable is used and is updated each time a loop is entered.

Example problems: The Rank, Golden Plate, Sum of Digits, Bear and Big Brother

6. Counter

At times, you would want to count the number of times you did something. Usually, a counter variable is used which is incremented you iterate.

Example problems: Bear and Big Brother

7. Math based

In some scenarios, one has to understand the math behind the solution for a problem, come up with the correct formula and program the formula.

Example problems: Frog Jumping, Integer Sequence Dividing, Definite Game, Vova and Train, Petya and Origami, King's Race, Vasya and Chocolate, Make a Triangle, Coins

8. Input size

Use of appropriate data type is important for correct computation avoiding overflow. For large input size, use of long long or long double is preferred.

Example Problems: Frog Jumping, King's Race, Vasya and Chocolates