GeeksforGeeks

Essential Maths for Competitive Programming

Detailed Course Syllabus

• Module 1

a. **Time Complexity**
   i. Analysis of Algorithms
   ii. Analysis of common loops, recursion
   iii. Asymptotic Notation
   iv. Space Complexity
   v. Order of Growth

b. **Basics**
   i. Arithmetic and Geometric Progressions
   ii. Solving First Competitive Programming Problem
   iii. Ranges of Input and Output
   iv. C++ Tricks for Competitive Programming
   v. Using Typedef and Macros in C++
   vi. Sum of first N natural numbers
   vii. Sum of squares of first N natural number
   viii. Factorial of a Number
   ix. Trailing Zeros in Factorial
   x. Maximum Power Dividing Factorial
   xi. Practice Problems

c. **Fibonacci**
   i. Fibonacci Numbers and Implementation
   ii. Properties of Fibonacci Numbers
   iii. Fibonacci Divisibility and GCD
   iv. Duddeney's Cow
   v. Count Binary Strings With No two consecutive 1's
   vi. Count ways to reach n-th stair
   vii. Sum of Fibonacci and of Fibonacci Squares
   viii. Fibonacci Log N Implementation
   ix. Pisano's Period
   x. Fibonacci Mod M
   xi. Practice Problems

d. **Bit Masking**
   i. Bitwise OR, AND, XOR
   ii. Bitwise Tricks
   iii. Count Set Bit
   iv. Check Kth bit is set or not
   v. Power of 2
   vi. One odd Occurring

vii. Two Odd Occurring
viii. Gray Code
ix. Power Set Using Bitwise
x. Practice Problems

• Module 2

a. **Prime Numbers**
   i. Prime Number and Concept
   ii. Sieve of Erathoneses
   iii. Segmented Sieve
   iv. Practice Problems
b. **Prime Factorization**
   i. All prime factors of a number
   ii. Least Prime factor concept
   iii. Prime factors for multiple queries
   iv. Practice Problems
c. **DIvisors**
   i. O(sqrt(N)) approach
   ii. Count Divisors
   iii. Numbers with Exactly 3 Divisors
   iv. Count Factors for Multiple queries
   v. Sum of Divisors and Multiple Queries
   vi. Practice Problems

• Module 3

a. **GCD & LCM**
   i. GCD & LCM Concept
   ii. Basic & Euclidean Algorithm
   iii. Bezout's Identity
   iv. Linear Diophantine Equation
   v. Practice Problems
b. **Number Theoretic Functions**
   i. Euler Totient
   ii. Properties of Euler Totient function
   iii. Euler Totient for 1 to N
   iv. Practice Problems

c. **Mathematics Principles**
    i. Inclusion & Exclusion Principle
    ii. Pigeon Hole Principle
    iii. Derangements
    iv. Counting Derangements
    v. Subarray sum divisible by Size
    vi. Practice Problems

• Module 4

a. **Modular Operations**
    i. Modular arithmetic Introduction
    ii. Modular Arithmetic in Competitive Coding
    iii. Range of int Type
    iv. Modular Arithmetic Properties
    v. Mod Inverse Concept
    vi. Mod Inverse Using Euler's Theorem
    vii. Working of Euler's Theorem
    viii. Mod Inverse Using Fermat Little
    ix. Mod Inverse from 1 to N
    x. Chinese Remainder Theorem
    xi. CRT Efficient Implementation
b. **Modular Exponentiation**
    i. Computing Power
    ii. Iterative Power
    iii. Matrix Exponentiation
    iv. Fibonacci Number using Matrix Exponentiation
c. **Binomial Concepts**
    i. Permutation
    ii. Printing All Permutations
    iii. Combination
    iv. nCr (Simple)
    v. nCr (recursive)
    vi. nCr MOD p for large Prime
    vii. Practice Problems
d. **Catalan Number**
    i. Catalan Number and Concept
    ii. Count Way to reach grid top
    iii. Number of Binary Trees

iv. Counting Polygon Triangulation
v. More Problems on Catalan
vi. Catalan Number Implementation Based Problems

• Module 5

a. Combinatorial Game Theory

i. Introduction
ii. Impartial Game Examples
iii. Grundy Number & its implementation
iv. Mex & Grundy Number
v. Composites Game
vi. Sprague Grundy Theorem
vii. The Game of Nim
viii. Working of Nim-Sum & Grundy Number Theorem
ix. Practice Problems

b. Geometric Algorithms

i. Orientation of three ordered points
ii. Line segment Intersection
iii. Convex Hull Problem
iv. Gift Wrapping Algorithm (or Janvi's March)
v. CPP Implementation of Jarvi's March algorithm
vi. Collinear Points in Jarvi's algorithm
vii. Graham Scan Algorithm for Convex Hull
viii. Graham Scan Algorithm in C++
ix. Closest Pair of Points
x. Implementation of Closest Pair in C++
xi. Working of Closest Pair Algorithm
xii. Sweep Line Algorithm for Any Two Intersection
xiii. Practice Problems.

c. Misc. Problems