Temario de Programación Competitiva en C++

Sesión 1: Introducción y Primeros Pasos (2h)

- Qué es la programación competitiva
- Plataformas: Codeforces, AtCoder, CodeChef, LeetCode
- Estructura de un problema competitivo
- Entrada/Salida básica con `cin` y `cout`
- Primer programa: "Hello World" y problemas A+B
- Plantilla básica de código

Sesión 2: Variables y Tipos de Datos (2h)

- Tipos primitivos: `int`, `long long`, `double`, `char`, `bool`
- Cuándo usar `long long` vs `int`
- Límites de tipos de datos y overflow
- Operadores aritméticos, lógicos y relacionales
- `typedef` y `using` para alias
- Problemas prácticos simples

Sesión 3: Estructuras de Control - Condicionales (2h)

- `if`, `else if`, `else`
- Operadores lógicos: `&&`, `||`, `!`
- `switch-case`
- Operador ternario
- Problemas de decisión
- Debugging básico

Sesión 4: For simples y patrones de iteración (2h)

- Sintaxis básica de `for` en C++
- Iterar rangos: índices y límites
- Patron común: acumuladores, contadores, búsqueda lineal con `for`
- Uso combinado con `if` para validaciones dentro del bucle
- Errores comunes (off-by-one)
- Problemas prácticos centrados en `for` simples (sumas, conteos, transformaciones)

Sesión 5: For anidados y recorridos estructurados (2h)

- Concepto de `for` anidado
- Recorrido de matrices (filas x columnas)
- Complejidad en anidamientos y análisis O(n^2), O(n^3)
- Patrones: recorridos por submatrices, ventanas 2D, diagonales
- Optimización y cuándo evitar profundos anidamientos
- Ejercicios: transposición, conteos en cuadrícula, búsqueda en matriz

Sesión 6: Arreglos Unidimensionales (2h)

- Declaración e inicialización
- Acceso y modificación de elementos
- Recorrido de arreglos
- Operaciones básicas: suma, promedio, búsqueda
- Problemas de arreglos básicos
- Diferencia entre arreglos estáticos y dinámicos

Sesión 7: Arreglos Multidimensionales (2h)

- Matrices (arreglos 2D)
- Declaración e inicialización de matrices
- Recorrido por filas y columnas
- Operaciones comunes con matrices
- Problemas de matrices
- Matrices 3D (introducción)

Sesión 8: Strings (2h)

- `string` de C++ STL
- Operaciones básicas: concatenación, longitud, acceso
- Comparación de strings
- Métodos útiles: `substr`, `find`, `replace`
- Conversión entre tipos (`to_string`, `stoi`)
- Problemas con strings

Sesión 9: Funciones (2h)

- Declaración y definición de funciones
- Parámetros por valor y por referencia
- Retorno de valores
- Funciones útiles en competitiva
- Sobrecarga de funciones
- Scope de variables

Sesión 10: Recursión (2h)

- Concepto de recursión
- Caso base y caso recursivo
- Stack de llamadas
- Factorial, Fibonacci, potencias
- Backtracking básico
- Cuándo usar recursión vs iteración

Sesión 12: Vectores (completo) (2h)

- `vector<T>` vs arreglos estáticos
- Declaración e inicialización
- Operaciones: `push_back`, `pop_back`, `size`, `clear`
- Acceso a elementos
- Iteración sobre vectores
- Vectores 2D
- Operaciones avanzadas: `insert`, `erase`, `resize`
- `vector<bool>` y sus peculiaridades
- Iteradores básicos
- Problemas básicos e intermedios con vectores

Sesión 13: Ordenamiento (2h)

- `sort()` de la STL
- Ordenamiento ascendente y descendente
- Comparadores personalizados
- `reverse()`
- `stable_sort()`
- Problemas que requieren ordenamiento

Sesión 15: Complejidad Algorítmica (2h)

- Notación Big O
- Complejidades comunes: O(1), O(log n), O(n), O(n log n), $O(n^2)$
- Cómo calcular complejidad
- Estimar si una solución pasa los límites de tiempo
- Análisis de casos peor, mejor y promedio
- Ejercicios de análisis

Sesión 16: Búsqueda Lineal y Binaria (2h)

- Búsqueda lineal: implementación y complejidad
- Búsqueda binaria: concepto y condiciones
- Implementación de binary search
- `lower_bound` y `upper_bound`
- Problemas clásicos de búsqueda binaria
- Binary search en respuesta

Sesión 17: Técnica de Dos Punteros (2h)

- Concepto de dos punteros
- Patrones comunes
- Problemas clásicos (suma de pares, subsecuencias)
- Ventana deslizante (sliding window)
- Optimización de $O(n^2)$ a O(n)

- Ejercicios prácticos

Sesión 18: Set (2h)

- `set<T>` y sus propiedades
- Operaciones: `insert`, `erase`, `find`, `count`
- Iteración sobre sets
- `multiset` para elementos duplicados
- `unordered_set` vs `set`
- Problemas usando sets

Sesión 19: Map (2h)

- `map<K, V>` para mapeo clave-valor
- Operaciones básicas
- Conteo de frecuencias
- `map` vs `unordered_map`
- `multimap`
- Problemas de conteo y mapeo

Sesión 20: Stack y Queue (2h)

- `stack<T>`: concepto LIFO
- Operaciones: `push`, `pop`, `top`
- `queue<T>`: concepto FIFO
- Operaciones de queue
- Problemas clásicos con pilas y colas
- Aplicaciones prácticas

Sesión 21: Priority Queue (2h)

- `priority_queue<T>` y heaps
- Max heap y min heap
- Operaciones: `push`, `pop`, `top`
- Comparadores personalizados
- Problemas que usan priority queue
- Simulación de eventos

Sesión 22: Deque (2h)

- `deque<T>` (double-ended queue)
- Operaciones en ambos extremos
- Cuándo usar deque vs vector
- Sliding window máximo/mínimo
- Problemas prácticos

Sesión 23: Complejidad Algorítmica (2h)

- Notación Big O
- Complejidades comunes: O(1), O(log n), O(n), O(n log n), O(n^2)
- Cómo calcular complejidad
- Estimar si una solución pasa los límites de tiempo
- Análisis de casos peor, mejor y promedio
- Ejercicios de análisis

Sesión 24: Algoritmos Greedy (2h)

- Concepto de algoritmos greedy
- Estrategia de elección local óptima
- Problemas clásicos greedy
- Cuándo funciona greedy y cuándo no
- Demostración de correctitud
- Ejercicios de greedy

Sesión 25: Prefix Sum y Diferencias (2h)

- Sumas acumulativas (prefix sum)
- Consultas de rango en O(1)
- Suma de subarreglos
- Difference array
- Aplicaciones prácticas
- Problemas de rangos

Sesión 26: Aritmética Modular (2h)

- Operador módulo `%`
- Propiedades de la aritmética modular
- MOD en sumas, restas, multiplicaciones
- Exponenciación modular
- Inverso modular (introducción)
- Problemas con MOD

Sesión 27: Números Primos (2h)

- Verificar si un número es primo (O(raiz n))
- Criba de Eratóstenes
- Factorización prima
- Divisores de un número
- Máximo común divisor (GCD) y mínimo común múltiplo (LCM)
- Problemas con primos

Sesión 28: Combinatoria Básica (2h)

- Permutaciones y combinaciones
- Coeficientes binomiales
- Triángulo de Pascal
- Cálculo eficiente de factoriales
- Principio de multiplicación y suma
- Problemas de conteo

Sesión 29: Bits y Operaciones Bitwise (2h)

- Representación binaria
- Operadores: `&`, `|`, `^`, `~`, `<<`, `>>`
- Máscaras de bits
- Contar bits activados
- Subconjuntos con bits
- Trucos útiles con bits

Sesión 30: Estrategias de Resolución (2h)

- Cómo leer y entender problemas
- Identificar patrones comunes
- Dividir el problema en subproblemas
- Casos de prueba y casos borde
- Debugging efectivo
- Gestión del tiempo en contests

Sesión 31: Contest Simulado y Repaso (2h)

- Contest completo estilo Codeforces Div. 3/4
- Resolución de 4-5 problemas
- Análisis de soluciones
- Revisión de conceptos clave
- Recursos para seguir aprendiendo
- Próximos pasos en programación competitiva