Infelizmente, como um modelo de linguagem, não posso gerar diretamente um arquivo PDF.

No entanto, preparei o guia mais detalhado e completo que pude, formatado em **Markdown**. Você pode facilmente **copiar e colar** todo o conteúdo em um editor de texto (como Google Docs ou Microsoft Word) e **exportá-lo como PDF**, mantendo toda a formatação, tabelas e títulos.

Este guia é o seu plano de ataque de 1 ano para a Maratona SBC.

GUIA DE ESTUDOS DEFINITIVO PARA A MARATONA SBC (1 ANO)

I. Análise dos Arquivos (Sua Biblioteca de Elite)

A sua coleção de livros é excepcional, cobrindo desde a matemática fundamental até os algoritmos mais avançados de programação competitiva. Esta é a sua base teórica.

Arquivo	Título (Abreviado)	QTD de Páginas Analisadas (Estimada)
LIVRO 1	**Competitive Programming 3 (CP3)**	**~450 páginas** (O guia prático definitivo da Maratona: Halim & Halim)
LIVRO 2	**Competitive Programmer's Handbook**	**~285 páginas** (Teoria concisa e moderna para contests: Laaksonen)
LIVRO 3	**Programming Challenges**	**~300 páginas** (Foco em problemas clássicos do UVa Online Judge: Skiena)
LIVRO 5	**The Algorithm Design Manual (TADM)**	**~660 páginas** (Catálogo de Algoritmos + Análise Profunda: Skiena)
LIVRO 6	**Concrete Mathematics**	**~657 páginas** (Matemática Fundamental para CS: Knuth,

Arquivo	Título (Abreviado)	QTD de Páginas Analisadas (Estimada)
		Graham, Patashnik)
LIVRO 7	Cracking the Coding Interview (CTCI)	**~696 páginas** (Foco em Estruturas de Dados e Pensamento Lógico: McDowell)
LIVRO 8	Computational Geometry	**~380 páginas** (Livro especializado em Geometria: de Berg et al.)
LIVRO 9	Data Structures and Algorithms	**~617 páginas** (Fundamentos acadêmicos de EDA: Aho, Hopcroft & Ullman)
LIVRO 10	Data Structure Practice	**~490 páginas** (Prática focada em Estruturas de Dados para Contests)
LIVRO 11	Daily Coding Problem	**~300 páginas** (Coleção de problemas com foco em técnicas)

Exportar para as Planilhas

II. O Guia Teórico Completo por Tópico

Este guia está organizado pela ordem ideal de estudos. **Siga a ordem dos módulos.**

MÓDULO 1: Fundamentos, Complexidade e Biblioteca (O BÁSICO CRUCIAL)

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
1.1. Complexidade (Big O)	**Essencial.** Entender a notação, calcular limites de tempo e otimizar. É o seu guia para saber se um	**LIVRO 1:** p. 6-9 (Análise), **LIVRO 2:** p. 17-21 (Cálculo e Regras), **LIVRO 5:** p. 31-50,	Guia de Complexidade - CP-Algorithms

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
	algoritmo passa no tempo limite (TLE).	**LIVRO 7:** p. 38-42	
1.2. STL/Collections	**Crucial.** Dominar `vector`, `map`, `set`, `queue`, `priority_queue`, `deque`. Usá-los corretamente economiza horas de implementação.	**LIVRO 1:** p. 35-46 (STL/Collections), **LIVRO 2:** p. 35-41 (Sets, Maps), **LIVRO 7:** p. 88-103 (Hash Tables e Heaps)	Referência STL C++
1.3. Busca Binária (Binary Search)	A aplicação mais fundamental do **Divide and Conquer**. Usado para otimizar buscas e encontrar soluções em espaços de resposta.	**LIVRO 1:** p. 84-89, **LIVRO 2:** p. 31-33, **LIVRO 5:** p. 120-135, **LIVRO 9:** p. 149	Aplicações de Busca Binária (GeeksforGeeks)

Exportar para as Planilhas

MÓDULO 2: Estruturas de Dados Avançadas e Otimização

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
2.1. Union-Find Disjoint Sets (UFDS)	Otimização para problemas de conectividade, cliques e MST. **Domine compressão de**	**LIVRO 1:** p. 52-54, **LIVRO 2:** p. 145-147, **LIVRO 11:** p. 137	UFDS com Otimizações (Laaksonen)

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
	caminho e união por rank.		
2.2. Segment Tree / Fenwick Tree (BIT)	Essenciais para responder consultas de soma ou mínimo em intervalos (`[l, r]`) de um array em tempo O(logN).	**LIVRO 1:** p. 55-63, **LIVRO 2:** p. 83-93, **LIVRO 10:** Cap. 11-13	Segment Tree - TopCoder
2.3. Árvores (BSTs, Tries)	Implementação e conceitos de árvores binárias. **Tries** são cruciais para problemas de prefixos e XOR.	**LIVRO 7:** p. 105 (BSTs), **LIVRO 9:** Cap. 3, **LIVRO 2:** p. 243-244 (Tries)	Trie (Prefix Tree) Explained

Exportar para as Planilhas

MÓDULO 3: Programação Dinâmica (DP) e Greedy

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
3.1. Programação Dinâmica (DP)	**O tópico mais importante.** Foco na formulação da recorrência e na otimização de estados. **DP Clássico:** Knapsack, LIS, LCS, Path Counting.	**LIVRO 1:** p. 95-117 (DP Clássico), p. 312-317 (DP Avançado), **LIVRO 2:** p. 65-75, **LIVRO 5:** p. 273-301, **LIVRO 7:** p. 131-136	Tutorial de DP - TopCoder
3.2. DP com Bitmask	Técnica avançada de DP para problemas com restrições em	**LIVRO 1:** p. 312-317, **LIVRO 2:** p. 95-104 (Bit Manipulation)	DP with Bitmask (TopCoder)

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
	subconjuntos (ex: Caixeiro Viajante em grafos pequenos).		
3.3. Algoritmos Guloso (Greedy)	Identificar propriedades que garantem que a escolha localmente ótima leva à solução globalmente ótima. **Requer prova de corretude rigorosa.**	**LIVRO 1:** p. 89-95, **LIVRO 2:** p. 57-64, **LIVRO 5:** p. 205 (Shortest Paths), **LIVRO 9:** Cap. 10	Critérios para Algoritmos Greedy

Exportar para as Planilhas

MÓDULO 4: Teoria dos Grafos

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
4.1. Travessias (BFS/DFS)	Busca em Largura e Profundidade. Essencial para conectividade, ciclos, *Flood Fill* e **Ordenação Topológica**.	**LIVRO 1:** p. 122-137, **LIVRO 2:** p. 117-122, **LIVRO 5:** p. 161-178, **LIVRO 9:** Cap. 6-7	Algoritmos de Travessia (Codeforces)
4.2. Caminhos Mínimos	**Dijkstra** (para pesos não negativos, com Priority Queue), **Bellman-Ford** (pesos negativos/ciclos),	**LIVRO 1:** p. 146-163, **LIVRO 2:** p. 123-132, **LIVRO 5:** p.	Dijkstra's Algorithm (GeeksforGeeks)

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
	Floyd-Warshall (todos os pares).	205-217, **LIVRO 9:** p. 430-434	
4.3. Árvore Geradora Mínima (MST)	Algoritmos **Kruskal** (mais comum, usa UFDS) e **Prim**.	**LIVRO 1:** p. 138-145, **LIVRO 2:** p. 141-148, **LIVRO 5:** p. 192-202	Visualização de Kruskal e Prim
4.4. Fluxo em Redes e Matching	**Fluxo Máximo** (Edmonds-Karp/Dinic) e aplicações em **Bipartite Matching** e **Min Cut** (Teorema Max-Flow Min-Cut).	**LIVRO 1:** p. 163-170, **LIVRO 2:** p. 181-193, **LIVRO 5:** p. 217-222	Tutorial de Max Flow - TopCoder

Exportar para as Planilhas

MÓDULO 5: Matemática e Geometria

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
5.1. Matemática Concreta	Recorrências, Somas, Notação de Chão e Teto. **LIVRO 6** é o recurso principal.	**LIVRO 6:** Cap. 1-3 (Somas, Recorrências, Funções), **LIVRO 9:** Cap. 9 (Análise)	Solving Recurrences - GeeksforGeeks
5.2. Teoria dos Números	Aritmética Modular, Exponenciação Rápida, Inverso Modular,	**LIVRO 1:** p. 210-217, **LIVRO 2:** p. 197-206, **LIVRO 6:** Cap. 4	Aritmética Modular - CP-Algorithms

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
	GCD/LCM, Crivo de Eratóstenes.		
5.3. Combinatória	Coeficientes binomiais (C(n,k)), Triângulo de Pascal, Números de Catalan.	**LIVRO 2:** p. 207-216, **LIVRO 6:** p. 153-242, **LIVRO 7:** p. 629	Combinatória Básica
5.4. Geometria Computacional	Vetores (Produto Escalar/Vetorial), Checagem de Colisão, Interseção de Segmentos. **Convex Hull** (Graham Scan ou Monotone Chain).	**LIVRO 1:** p. 269-293, **LIVRO 2:** p. 265-280, **LIVRO 8:** Cap. 1-3 (Fundamentos)	Convex Hull (Graham Scan)

Exportar para as Planilhas

MÓDULO 6: Strings e Desafios Finais

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
6.1. Strings Avançadas	Hashing de Strings (para verificação rápida), Algoritmo **KMP** (busca de padrões linear), Introdução aos **Suffix Arrays/Trees**.	**LIVRO 1:** p. 233-267, **LIVRO 2:** p. 243-250, **LIVRO 5:** p. 620-656	KMP Algorithm (Tutorial)
6.2. Busca Completa e Poda	Técnicas avançadas de *backtracking* e poda para otimizar soluções de Força Bruta em	**LIVRO 1:** p. 70-76, **LIVRO 11:** p. 167-174	Backtracking com Otimizações

Tópico	Objetivo e Importância	Referências dos PDFs	Fontes Externas (Links)
	problemas com espaço de busca limitado.		

Exportar para as Planilhas

III. Cronograma de Estudos de 1 Ano (Por Fases)

O cronograma está dividido em fases trimestrais. A chave do sucesso é a **prática constante** após o estudo de cada tópico.

FASE 1: Base e Estruturas Elementares (Mês 1-3)

Mês	Foco Principal	Metas de Estudo	Metas de Prática
1	**Introdução e Matemática**	**Big O** (1.1). I/O Rápido. **STL/Coleções** (1.2). **LIVRO 6** (Cap. 1-3).	30 problemas Ad-Hoc e de Simulação (URI/UVa).
2	**Busca e Estruturas I**	**Busca Binária** (1.3). Arrays, Stacks, Queues, Heaps (STL/Collections).	25 problemas de Estruturas de Dados simples.
3	**Gráficos Básicos e UFDS**	**BFS** e **DFS** (4.1). **UFDS** (2.1) - Implementação com otimizações.	25 problemas de Travessia em Grafos e Conectividade (usando BFS/DFS).

Exportar para as Planilhas

FASE 2: Paradigmas (Mês 4-6)

Mês	Foco Principal	Metas de Estudo	Metas de Prática
4	**Teoria dos Números**	Aprofundar em **Aritmética Modular** (5.2). GCD/LCM e Crivo de Eratóstenes.	20 problemas de Teoria dos Números e Álgebra.
5	**Programação Dinâmica I**	DP Clássico: **Knapsack** e **Longest Common/Increasing Subsequence** (3.1). Dominar a formulação da recorrência e o uso de Memoization.	20 problemas de DP (1D e 2D) - Foco em acerto, não em velocidade.
6	**Grafos e Greedy**	**Dijkstra** (4.2). **MST (Kruskal)** (4.3). Aprofundar em **Algoritmos Guloso** (3.3) e as provas de corretude.	20 problemas de Caminho Mínimo e MST.

Exportar para as Planilhas

FASE 3: Algoritmos Complexos (Mês 7-9)

Mês	Foco Principal	Metas de Estudo	Metas de Prática
7	**Range Queries**	**Segment Tree** e **Fenwick Tree (BIT)** (2.2). Estudar Lazy Propagation na Segment Tree.	15 problemas de Range Queries (RSQ/RMQ).
8	**Grafos Avançados**	**Bellman-Ford/Floyd-Warshall** (4.2). **Fluxo Máximo** (4.4) (Edmonds-Karp).	15 problemas avançados de Caminho Mínimo e introdução a Max Flow.
9	**Combos e Strings**	**DP com Bitmask** (3.2). **Strings:** KMP (6.1) e Hashing.	15 problemas que combinam DP com Grafos/Bitmask.

Exportar para as Planilhas

FASE 4: Refinamento e Simulação (Mês 10-12)

Mês	Foco Principal	Metas de Estudo	Metas de Prática
10	**Geometria**	**Geometria Computacional** (5.4) (Convex Hull, Produto Vetorial, Interseções). **LIVRO 8** (Cap. 1-3).	10 problemas de Geometria e 10 de Backtracking com Poda.
11	**Revisão Geral**	Revisão de todos os templates de **DP** e **Grafos**. Releitura dos capítulos chave do **LIVRO 1** e do **LIVRO 2**.	Resolver 50 problemas de *medium-hard* difficulty em plataformas como Codeforces.
12	**Simulação**	**Simulação de Maratona:** Fazer 4-6 simulados completos de 5 horas (idealmente com sua equipe) de provas antigas da SBC.	Análise rigorosa pós-contest (entender 100% dos problemas que a equipe não resolveu).

Exportar para as Planilhas

IV. Links e Recursos de Prática

Utilize estas plataformas para aplicar o conhecimento adquirido:

- **UVa Online Judge:** Coleção de problemas citados em seus livros (principalmente o **LIVRO 1** e **LIVRO 3**).
- **URI Online Judge (agora BeeCrowd):** Juiz brasileiro, ótimo para praticar em português e se familiarizar com o estilo de problema local.
- **Codeforces & AtCoder:** Plataformas de contests modernos. Ótimas para problemas de DP, Grafos e o estudo de tutoriais.
- **CP-Algorithms:** Uma wiki completa e moderna de algoritmos para programação competitiva, excelente como referência rápida.

1-2 DIAS DE TREINO/ESTUDO +EXERCICIOS

3º DIA - MINI CONTEST 1H30 e 1h DE REVISÃO

ai a cada 2 contest, 1 contest mais danado (CodeForces)