

# Maratona de Programação

Paulo Cezar Pereira Costa  
Humberto Longo

Instituto de Informática  
Universidade Federal de Goiás

26 de novembro de 2012

# Introdução

## O que são?

- São competições onde um programador, ou grupo de programadores, deve apresentar soluções computacionais corretas para um dado conjunto de problemas.
  - A correção é automatizada e realizada por meio de testes:
    - um conjunto de dados de entrada.
    - um meio de validar a resposta gerada.
  - O programa solução deve:
    - gerar um conjunto de dados de saída correspondente ao conjunto de dados recebido na entrada.
    - respeitar limites de tempo de execução e de quantidade de memória usada.
    - o resultado do programa deve ser validado corretamente nos testes realizados.

## Por que é Bom?

- Essas competições estimulam a capacidade de resolver problemas computacionais rápida e eficientemente, uma das principais habilidades exigidas de um profissional de Computação.
- Estimulam o raciocínio lógico, o que acaba resultando em um melhor desempenho acadêmico.
- Você busca conhecer todos os recursos das linguagens de programação.
- Você ganha motivação para estudar algoritmos e estruturas de dados.
- Boas colocações nessas competições enriquecem seu currículo como programador e resolvido de problemas, o que é muito valorizado por grandes empresas da área.
- Possibilidade de viagem para vários destinos do Brasil e do mundo.
- É sempre bom competir!

## Como funcionam?

- Variam um pouco de estilo e regras, mas em todas é necessário programar.
- Existe um tempo fixo para resolver o conjunto de problemas propostos.
- Geralmente, para ganhar, você deve resolver o maior número de problemas no menor tempo possível.

## Algumas competições

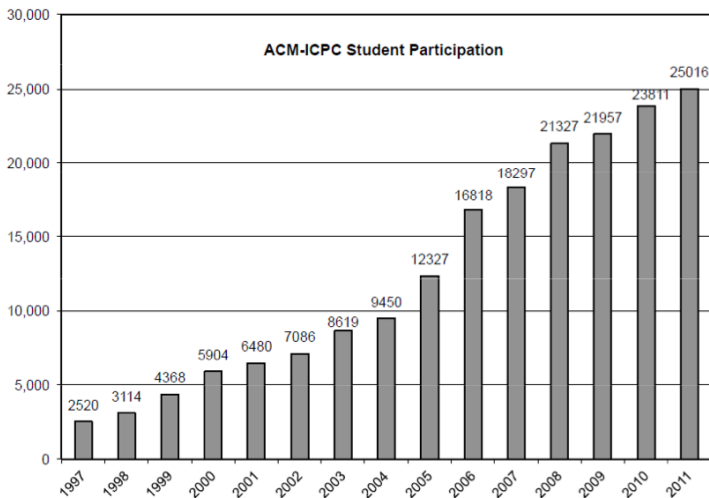
- Ensino médio:
  - Olimpíada Brasileira de Informática (OBI)
  - International Olympiad in Informatics (IOI)
  - TopCoder High School Tournament (TCHS)
- Ensino superior:
  - Maratona de Programação
  - ACM International Collegiate Programming Contest (ACM-ICPC)
  - TopCoder Collegiate Challenge (TCCC)
- Livre:
  - TopCoder Open (TCO)
  - Google Code Jam (GCJ)
  - Facebook Hacker Cup
  - Internet Problem Solving Contest (IPSC)
  - $\infty$

# Maratona de Programação

- *International Collegiate Programming Contest*
- Competição de programação existente desde 1970 e organizada pela ACM desde 1977.
- Patrocinada pela IBM.
- A missão do ICPC é dar oportunidades aos estudantes para interagir com alunos de outras universidades e demonstrar sua capacidade de **resolver problemas, programar e trabalhar em grupo**.



- A competição possui duas etapas
  - Finais Regionais – realizadas localmente, em vários países ao redor do mundo.
  - Final Mundial – reúne os melhores colocados das competições regionais.
- A etapa regional ocorre no ano anterior à final mundial
  - Os times se classificam nas regionais de 2012 para competir a final mundial de 2013.
- Apenas um time por instituição é classificado para a final mundial.



**Evolução da Competição – Número de Estudantes (ACM ICPC fact Sheet 2012)**

- 2007/2008: mais de 6700 times de 1821 escolas de 83 países.
  - 100 participaram da Final Mundial do evento, em Banff, Canadá.
  - Participação de 4 times brasileiros (IME, ITA, IME-USP, Unicamp) na final mundial.
- 2011/2012: mais de 8338 times (mais de 25000 estudantes) de 2219 instituições de 88 países competiram em regionais nos 6 continentes.
  - 112 times na Final Mundial do evento, em Varsóvia, Polônia.
  - 6 times brasileiros (ITA,UFCG,UFPR,UFPE, UFRJ e IME-USP) participaram da final.

## ICPC no Brasil

- Maratona de Programação desde 1996.
- Realizada pela SBC desde o ano 2000.
- Apoio do CNPq desde 2002.
- Realizada em parceria com a Fundação Carlos Chagas desde o ano 2006.



# Maratona de Programação

## Competição em duas fases

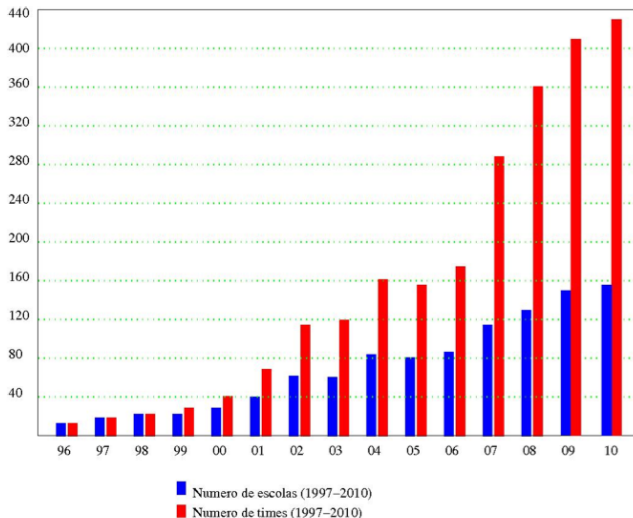
**Regional** Classificatória para a final nacional.

**Nacional** Final brasileira e classificatória para a final mundial.

Na América do Sul existem três regionais do ICPC.

- South America/Brazil Regional – A Final Nacional da Maratona de Programação.
  - South America/North Regional – Colômbia, Equador, Panamá, Venezuela.
  - South America/South Regional – Argentina, Bolívia, Chile, Peru, Paraguai, Uruguai.
- 
- <http://maratona.ime.usp.br/>
  - <http://www.inf.ufg.br/maratona/>
  - <https://www.facebook.com/maratonago>

## Crescimento da Maratona



## Como foi em 2011

Região	times		escolas		final	
Centro-Oeste	25	4.7%	9	4.7%	2	4.0%
Nordeste	116	21.6%	40	20.9%	11	22.0%
Norte	49	9.1%	16	8.4%	4	8.0%
Sudeste	256	47.8%	91	47.6%	25	50.0%
Sul	90	16.8%	35	18.3%	8	16.0%
<b>Total</b>	<b>536</b>		<b>191</b>		<b>50</b>	

## Participação do INF - Fase Regional

Ano	Local	Equipes	Instituições	Classificação
2007	Brasília-DF	5	3	3º
2008	Goiânia-GO	7	3	1º e 3º
2009	Goiânia-GO	11	4	1º, 3º e 4º
2010	Goiânia-GO	10	3	1º, 4º, 6º e 8º
2011	Goiânia-GO	14	4+1	1º, 2º, 5º e 8º
2012	Goiânia-GO	10	3+1	1º, 2º, 3º, 6º e 8º



## Participação do INF - Final Brasileira

Ano	Local	Equipes	Instituições	Classificação
2008	Vila Velha-ES	51/360	129	28º
2009	Campinas-SP	52/410	145	22º
2010	Joinville-SC	51/432	157	20º
2011	Goiânia-GO	50/536	191	11º
2012	Londrina-PR	50/545	194	13º

## Como funciona?

- Competição voltada para alunos de cursos de Ciência da Computação, Engenharia de Computação e áreas afins.
- Os times são compostos de três estudantes com até 5 anos de estudos universitários ou 23 anos de idade.
  - Podem ter participado de no máximo 4 regionais e 1 final mundial.
- Os times recebem de 8 a 12 problemas computacionais para serem resolvidos durante a competição (5 horas).
- Quando um time considera que resolveu um problema submete aos juízes que, *online*, dizem se a solução está ou não correta.
- Uma solução correta resolve um conjunto de testes dos juízes, desconhecido dos alunos e recebe um balão.

## Como funciona?

- Você deve saber programar em C, C++ ou Java.
- Cada equipe tem acesso a um computador, sem conexão à internet, e apenas a material impresso.
- Objetivo é apresentar soluções computacionalmente corretas para um dado problema no menor tempo possível.
  - O time vencedor é aquele que resolver mais problemas durante as 5 horas de competição
  - Em caso de empate vence o time com a menor “penalidade”
    - Soma das penalidades dos problemas corretamente resolvidos;
    - A penalidade de um problema é dada pelo número de minutos decorridos desde o início da competição até o momento da primeira submissão correta;
    - Uma penalidade de 20 minutos é adicionada por cada submissão incorreta feita antes da primeira submissão correta.

## Como funciona?

- Cada problema contém:
  - Informações para contextualização (Background)
  - O enunciado do problema
  - Informações sobre a entrada (Input)
  - Informações sobre a saída (Output)
  - Exemplo de entrada (Sample Input)
  - Exemplo de saída (Sample Output)

## Como funciona?

- Os juízes possuem datasets que são utilizados para testar a solução submetida
- Esses datasets contém instâncias de testes bastante diferentes dos exemplos contidos nos problemas do caderno de questões
- Os juízes informarão uma das seguintes respostas a uma solução submetida por um time (sem nenhum detalhe adicional)
  - Yes
  - No - Wrong Answer (WA)
  - No - Presentation Error (PE)
  - No - Time Limit Exceeded (TLE)
  - No - Runtime Error (RE)
  - No - Compile Error (CE)
- O sistema não responde nada além disto. Não é possível saber, por exemplo, se um programa que recebeu TLE produz a saída correta, ou em qual linha está o erro de um programa que recebeu CE.

## E se eu me der bem?

- Os melhores colocados da primeira fase por sede avançam para a Final Nacional da Maratona de Programação.
- Medalhas aos 10 primeiros colocados na final nacional
  - Ouro para os três primeiros
  - Prata para os três seguintes
  - Bronze para os quatro últimos
- Cópia do Troféu “Maratona de Programação” para o vencedor
- Premiação para os melhores colocados de cada região
- Vencedor ganha vaga na Final Mundial do ICPC
  - *Outros  $N$  times melhores colocados podem ter direito a participar da Final Mundial, dependendo do nro. de vagas adicionais atribuídas ao Brasil.*

# Como se preparar

# Como Se Preparar?

Para estar bem preparado para participar dessas competições é necessário passar por varias etapas:

- ① Escolher uma linguagem;
  - Dominar sintaxe, comandos básicos e as funções mais usadas em competições.
- ② Aprender noções de Complexidade de Algoritmos;
  - Calcular complexidade de tempo e memória e estimar o tempo de execução;
- ③ Dominar as estruturas de dados básicas;
  - Vetores, strings, pilha, fila, etc.
- ④ Dominar Entrada e Saída;
  - Leitura dos diferentes tipos, formatação de saída.
- ⑤ Dominar algoritmos clássicos e técnicas comuns para resolução de problemas .



## Alguns exemplos de algoritmos clássicos e técnicas comuns

- Recursão, Backtracking
  - Saber enumerar permutações, combinações e arranjos.
- Grafos
  - Estruturas de dados para representá-los; Algoritmos de busca: BFS, DFS; Árvore Geradora Mínima; Caminhos mínimos.
- Programação Dinâmica
  - Entender a idéia e conhecer os problemas clássicos.
- Matemática
  - Conceitos de combinatória, probabilidade, cálculo e teoria dos números.
- Geometria / Geometria Computacional
- Algoritmos gulosos, divisão e conquista, estruturas de dados, ...

# Como Se Preparar?

## E como dominar essas técnicas?

- Estudando as mesmas em livros, artigos e tutoriais encontrados na internet;
- Praticando em Online Judges;
  - Existem várias páginas na internet onde é possível resolver problemas no estilo da maratona e conferir se sua solução está correta.
- Participando de competições online;
  - Vários Online Judges organizam competições para que os competidores possam treinar.
  - Sites como topCoder, Codeforces e Codechef organizam competições regularmente.
- Esclarecendo dúvidas e pedindo ajuda a competidores mais experientes.

## Material para estudo - Livros voltados para competições

- Competitive Programming 2: This increases the lower bound of Programming Contests. Again. - Steven Halim, Felix Halim
- Programming Challenges - Steven S. Skiena, Miguel A. Revilla
- The Art of Algorithms and Programming Contests. - Rujia Liu, Liang Huang
- Art of Programming Contest - Ahmed Shamsul Arefin

## Material para estudo - Referências para desenvolvimento de algoritmos

- Introduction to Algorithms. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest. MIT Press/MacGraw Hill, 1990.
- Introduction to Algorithms: A Creative Approach. Udi Manber. Addison-Wesley, 1989.
- Algorithms in C Parts 1-5. Robert Sedgewick. 3rd. Edition, vol. 1. Addison Wesley Longman, 1998.
- Computational geometry: An introduction. F.P. Preparata and M.I. Shamos. Texts and Monographs in Computer Science, Springer-Verlag, New York, 1985.

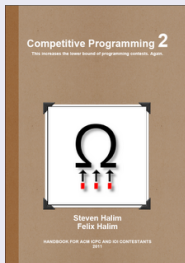
## Material para estudo - Referências para desenvolvimento de algoritmos

- Grafos e Algoritmos Computacionais. J. L. Szwarcfiter. Campus, Rio de Janeiro, 1986.
- Data Structures and Algorithms. Alfred V. Aho, Jhon E. Hopcroft and Jeffrey Ullman Addison Wesley, 1983.
- Concrete Mathematics. Donald E. Knuth, Ronald L. Graham and O. Patashnik. 2nd Edition Addison-Wesley, 1994.
- Computational Complexity. Papadimitriou, C.H., Addison-Wesley, 1993.

# Como Se Preparar?

## Material para estudo - Uma “bibliografia” indicada

- Competitive Programming 2 - Steven Halim, Felix Halim



<https://sites.google.com/site/stevenhalim/>

- [http://www.ime.usp.br/~pf/algoritmos\\_para\\_grafos/](http://www.ime.usp.br/~pf/algoritmos_para_grafos/)
- <http://www.cplusplus.com/reference/>

# Como Se Preparar?

## Online judges - A lista é extensa...

- URI Online Judge - <http://www.urionlinejudge.com.br/>
- UVa - <http://uva.onlinejudge.org/>
- Live Archive - <http://livearchive.onlinejudge.org/>
- SPOJ - <http://www.spoj.pl/>
- TJU - <http://acm.tju.edu.cn/toj/>
- SGU - <http://acm.sgu.ru/>
- PKU - <http://poj.org/>
- Timus - <http://acm.timus.ru/>
- ZOJ - <http://acm.zju.edu.cn/onlinejudge/>
- SPOJ BR - <http://br.spoj.pl/>
- USACO - <http://train.usaco.org/usacogate>
- ∞

# Como Se Preparar?

## Online judges - Por onde começar

- URI Online Judge - <http://www.urionlinejudge.com.br/>
  - Portal brasileiro bastante didático com problemas em português e inglês classificados de acordo com a dificuldade e a possível técnica envolvida na solução.
- SPOJ BR – <http://br.spoj.pl/>
  - Página com problemas em português, de seletivas, regionais e nacionais passadas. Seção com problemas da OBI pode ser bastante interessante para alunos iniciantes.
- USACO – <http://train.usaco.org/usacogate>
  - Curso de treinamento para os alunos dos Estados Unidos interessados em participar da olimpíada de informática. Dividido em seções, textos breves explicando técnicas e fornecendo mais referências e listas de problemas para serem resolvidos.



## Online judges - Partindo para problemas mais desafiadores

- SPOJ – <http://www.spoj.pl/>
  - Coleção vasta de problemas com enunciados em inglês de origens e dificuldades variadas.
- Live Archive – <http://livearchive.onlinejudge.org/>
  - Coleção de problemas das provas de regionais e finais mundiais passadas.
- UVa – <http://uva.onlinejudge.org/>

## Online judges - Ferramentas

- Virtual Online Contests – <http://ahmed-aly.com/voc/>
  - Permite criar placares para competições envolvendo problemas de vários juízes e acompanhar quais problemas os usuários resolveram. Também tem vários outros recursos, como buscar por problemas de acordo com a técnica envolvida na solução e criar times e grupos.
- uHunt – <http://uhunt.felix-halim.net/>
  - Ferramenta para o UVA online-judge que gera estatísticas dos problemas resolvidos por cada usuário e dá sugestões de quais problemas resolver. Criado por um dos autores do livro Competitive Programming facilita bastante encontrar e manter um histórico de quais dos problemas discutidos no livro já foram resolvidos.

## Competições Online

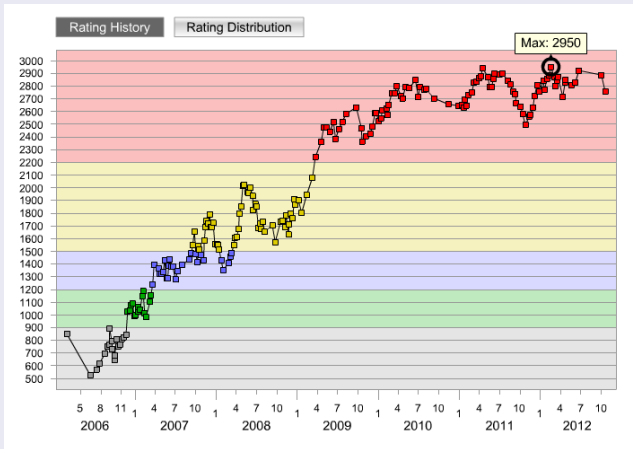
- TopCoder SRMs – <http://topcoder.com/tc>
- Codeforces – <http://codeforces.com/contests>
- Codechef – <http://codechef.com/>
- TopCoder Open – <http://community.topcoder.com/tco13/>
- Google Code Jam – <http://code.google.com/codejam/>
- Facebook Hacker Cup – <https://www.facebook.com/hackercup>
- IPSC – <http://ipsc.ksp.sk/>
- **Calendário de competições** – <http://codingdoor.com/>

## Competições Online – TopCoder

- TopCoder SRMs – <http://topcoder.com/tc>
  - Geralmente ocorrem de 15 em 15 dias.
  - Competidores recebem uma pontuação e são rankeados e separados em divisões. Div. 2 tem problemas mais simples, a medida que o competidor vai apresentando um bom desempenho sua pontuação aumenta e ele passa a competir na div. 1 onde os problemas são mais desafiadores.
  - Competição dividida em fases: Coding Phase, Challenge Phase e System Test.
  - Explicações das soluções para os problemas divulgadas algum tempo após a competição.

# Como Se Preparar?

## Competições Online – TopCoder



## Competições Online – Codeforces

- Codeforces – <http://codeforces.com/contests>
  - Competições semelhantes as do topCoder, porém com mais problemas e maior duração. Os competidores são separados em duas divisões e as competições tem duas etapas, Coding Phase e System Test.
  - Explicações das soluções para os problemas geralmente são divulgadas algum tempo após a competição.

## Competições Online – Codechef

- Codechef – <http://codechef.com/>
  - Dois tipos de competições, ambas sem distinção entre os competidores.
    - Long Contests – Competição mais didática e de longa duração, acontece todos os meses a partir do dia primeiro e tem duração de 10 dias. A prova é composta por 10 problemas de dificuldades variadas e o objetivo é que o competidor tenha tempo de pesquisar e aprender novas técnicas para resolver os problemas propostos.
    - Short Contests – Competições curtas, com 2,5 horas de duração que acontecem uma vez por mês. A prova tem 5 questões de dificuldades variadas.
  - Explicações das soluções sempre são divulgadas pouco tempo depois das competições.

## Entrar em contato com outros competidores

- Praticamente todos os Online Judges tem fóruns onde os usuários podem trocar informações e dicas sobre os problemas, é importante usar bastante essa ferramenta.
- <http://br.groups.yahoo.com/group/maratona/> – Grupo nacional da maratona no Yahoo Grupos, lugar onde competidores de todo o Brasil podem discutir sobre resoluções de problemas e competições de programação.
- <https://www.facebook.com/groups/maratonago/> – Grupo destinado a alunos, professores, ex-competidores e entusiastas da Maratona de Programação no estado de Goiás.



# Como Se Preparar?

## Mais alguns endereços ...

- <https://sites.google.com/site/obiufg/Principal> – Página do treinamento da OBI oferecido pelo INF-UFG, livro preparatório para OBI pode ser encontrado nesse link.
- <http://inf.ufg.br/~cc080153/> – Alguns materiais disponibilizados no Facebook também podem ser encontrados aqui.
- <http://www.ime.usp.br/~cassio/boca/> – site do BOCA

Dúvidas?