

MC721A – Desafios de Programação III

Primeiro semestre de 2015

Professor responsável:
Fábio Luiz Usberti (fusberti@ic.unicamp.br) – sala 15 (IC1).

Monitor:
Lucas Porto Maziero (PED) – lucasporto1992@gmail.com

1 Páginas da Disciplina

Página do Ensino Aberto:
<http://www.unicamp.br/ea/>

Wiki de Maratona do IC:
<http://www.ic.unicamp.br/~maratona/wiki/>

2 Horário das Aulas

Dia	Horário	Sala
Sextas-feiras	14 – 18	304 (IC3)

3 Ementa

Estruturas de dados. Busca por padrões. Ordenação. Combinatória. Teoria dos Números. Backtracking. Algoritmos em grafos. Programação dinâmica. Reticulados. Geometria computacional.

4 Programa

1. Introdução à programação competitiva.
2. Estruturas de dados e bibliotecas.
3. Busca exaustiva.
4. Divisão-e-conquista.
5. Algoritmos gulosos.
6. Programação dinâmica.
7. Grafos.
8. Matemática discreta.
9. Cadeias de caracteres.
10. Geometria computacional.

5 Critério de Avaliação

A avaliação da disciplina será composta por exercícios de programação. Em cada aula serão disponibilizados de 5 a 10 exercícios. Ao concluir um exercício, o aluno recebe uma pontuação, correspondente à data da submissão:

Submissão	Pontuação
Durante a aula em que o exercício foi disponibilizado	2
Em uma semana (até quinta-feira da semana seguinte)	1
Durante o semestre (até dia 2 de julho)	0.5
Não entregue	0

Cálculo da nota final

Sejam X_i e F_i a pontuação total e o número de faltas do i -ésimo aluno, respectivamente. A nota final NF_i do i -ésimo aluno será calculada como mostrado a seguir:

Caso 1: $X_i < 50$

$$NF_i = 5.0 \times \frac{X_i}{50} - 0.25 \times F_i$$

Caso 2: $X_i \geq 50$

$$NF_i = NF_{\min} + (NF_{\max} - NF_{\min}) \left(\frac{X_i - X_{\min}}{X_{\max} - X_{\min} + \delta} \right) - 0.25 \times F_i + 2.5 \times C_i$$

Onde:

- $X_{\min} = \min\{X\}$, considerando somente os alunos com $X_i \geq 50$.
- $X_{\max} = \max\{X\}$, considerando somente os alunos com $X_i \geq 50$.
- NF_{\min} , variável cujo valor depende de X_{\min} .
- NF_{\max} , variável cujo valor depende de X_{\max} .
- $\delta = 10^{-6}$, constante infinitesimal para evitar divisão por zero.
- $C_i \in [0, 1]$, nota resultante da avaliação do desempenho do aluno em competições on-line (Top-coder, Codeforces, Codechef, dentre outras).

Pontuação mínima	NF_{\min}
$50 \leq X_{\min} < 60$	5.0
$60 \leq X_{\min} < 70$	5.5
$70 \leq X_{\min} < 80$	6.0
$80 \leq X_{\min} < 90$	6.5
$90 \leq X_{\min} < 100$	7.0
$X_{\min} \geq 100$	7.5

Pontuação máxima	NF_{\max}
$50 \leq X_{\max} < 60$	5.4
$60 \leq X_{\max} < 70$	5.9
$70 \leq X_{\max} < 80$	6.4
$80 \leq X_{\max} < 90$	6.9
$90 \leq X_{\max} < 100$	7.4
$X_{\max} \geq 100$	7.5

Situação do aluno

- o aluno com frequência maior ou igual a 75% e nota final $NF_i \geq 5.0$ estará **aprovado**.
- o aluno com frequência maior ou igual a 75% e nota final $NF_i < 5.0$ estará **reprovado por nota**.
- o aluno com frequência inferior a 75% estará **reprovado por frequência**.

Observações Importantes

- **Não haverá exame** nesta disciplina.
- O critério de avaliação obriga que o aluno, para ser aprovado, tenha **pontuação maior ou igual a 50**. Isso corresponde a 25 exercícios resolvidos em aula; ou 50 exercícios resolvidos ao longo da semana; ou 100 exercícios resolvidos ao longo do semestre; ou alguma combinação desses.
- O **cálculo da nota final** é feito de tal forma que o aluno com a maior pontuação (e $X_i \geq 50$) receberá nota igual a NF_{\max} , desconsiderando penalidade por faltas. Já o aluno com a menor pontuação (e $X_i \geq 50$) receberá nota igual a NF_{\min} , desconsiderando penalidade por faltas.
- A presença do aluno será verificada **duas vezes por aula**: no início (14:00) e fim (18:00) de cada aula. Cada lista de presença comprova o comparecimento do aluno em duas horas-aula.
- Considerando que a presença será verificada 30 vezes ao longo do semestre, o aluno deverá assinar pelo menos 23 listas de presença para não ser **reprovado por falta**.
- Um dos objetivos desta disciplina consiste em preparar o aluno para que este tenha um bom desempenho nas maratonas de programação. Para isso, é essencial que o aluno seja **auto-didata**, consultando referências bibliográficas que permitam um maior aprofundamento dos assuntos tratados ao longo do curso.
- Qualquer tentativa de fraude considerada para fins de avaliação implicará em **nota final igual a zero** para todos os envolvidos, sem prejuízo de outras sanções previstas no regimento da universidade.
- Os exercícios devem ser feitos **individualmente**, portanto cópia de programas de outros alunos ou de terceiros será considerada uma tentativa de fraude.
- Como a correção dos exercícios é feita de modo automático, por meio de programas computacionais, **nenhuma nota atribuída a uma atividade será revista**. A única revisão possível será na contagem da frequência e deverá ser solicitado pelo aluno **até dia 02/07**.
- Antes do início de cada aula, começando às 13:00 (na própria sala de aula), haverá uma discussão de algum tópico de interesse da disciplina. A participação é opcional mas **altamente recomendada**.

6 Atendimento

Após as aulas teóricas, o professor estará disponível para esclarecimento de dúvidas. Para atendimento extra-classe, envie uma mensagem pelo ensino aberto para o monitor ou professor.

7 Bibliografia

1. S. Halim e F. Halim. Competitive Programming 2, Second Edition Lulu (www.lulu.com), 2011. (IMECC – 005.1 H139c)
2. S. S. Skiena, M. A. Revilla. Programming Challenges: The Programming Contest Training Manual, Springer, 2003.
3. T.H. Cormen, C.E. Leiserson, R.L.Rivest e C. Stein. Introduction to Algorithms. 2nd Edition, McGraw-Hill, 2001. (no. chamada IMECC – 005.133 Ar64j 3.ed.)
4. U. Manber. Introduction to Algorithms: A Creative Approach. Addison-Wesley. 1989. (no. chamada IMECC – 005.133 Ec53t 2.ed.)

Tabela 1: Calendário da disciplina.

Mês	Dia	Evento	Obs.
Fevereiro	27, sexta-feira	Início das aulas	Aula 1
Março	6, sexta-feira		Aula 2
Março	13, sexta-feira		Aula 3
Março	20, sexta-feira		Aula 4
Março	27, sexta-feira		Aula 5
Abril	3, sexta-feira	Não haverá atividades	
Abril	10, sexta-feira		Aula 6
Abril	17, sexta-feira		Aula 7
Abril	24, sexta-feira		Aula 8
Maiο	1, sexta-feira	Não haverá atividades	
Maiο	8, sexta-feira		Aula 9
Maiο	15, sexta-feira		Aula 10
Maiο	22, sexta-feira		Aula 11
Maiο	29, sexta-feira		Aula 12
Junho	5, sexta-feira	Não haverá atividades	
Junho	12, sexta-feira		Aula 13
Junho	19, sexta-feira		Aula 14
Junho	26, sexta-feira		Aula 15
Julho	3, sexta-feira	Divulgação de notas	