

**PROVA 1 DE ESTRUTURAS DE DADOS**  
**BCC, 1o. SEMESTRE DE 2012**

**Instruções:**

1. Não destaque as folhas do caderno de soluções.
  2. A prova pode ser feita a lápis. Cuidado com a legibilidade.
  3. Não é permitido o uso de folhas avulsas para rascunho.
  4. Não é necessário apagar rascunhos no caderno de soluções.
  5. Aserções imprecisas valem pouco. Justifique suas asserções (dentro do razoável!).
- 
1. [3 pontos]
    - (i) Dê uma descrição de alto nível das estruturas de dados e algoritmos que você implementou para resolver o EP1.
    - (ii) Faça uma discussão sobre a eficiência de sua solução do EP1. Quanto tempo seu programa demora para processar uma instância com  $N$  palavras? De quanto espaço seu programa precisa para processar uma instância com  $N$  palavras? Faça uma estimativa da memória necessária no caso em que a entrada tem 1GB. Justifique sua resposta.
    - (iii) Você conhece outras formas de se resolver o EP1? Como é a complexidade destas outras soluções? (Faça uma análise como em (ii).)
  2. [3 pontos] Um time com  $2N$  jogadores  $1, \dots, 2N$  passará uma semana concentrado, treinando. Cada dois deles compartilharão um quarto. Para cada par de jogadores, você sabe se eles concordam ou não dividir um quarto. O seu trabalho é descobrir quais são todas as formas possíveis de se organizar a divisão dos  $2N$  jogadores em  $N$  quartos; vamos chamar esse problema de o Problema da Divisão em Quartos (PDQ).
    - (i) Suponha que  $N = 4$  e os pares de jogadores que concordam em compartilhar um quarto são  $\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}, \{5, 6\}, \{6, 7\}, \{7, 8\}$ , e  $\{1, 8\}$ . Resolva esta instância (isto é, diga quais são todas as possíveis formas de colocar esses 8 jogadores em 4 quartos).
    - (ii) Defina (precisamente) o Problema da Cobertura Exata (PCE) e dê um exemplo.
    - (iii) Descreva (precisamente) uma redução do PDQ para o PCE. Dê uma instância para o PCE cujas coberturas exatas correspondem à solução do exemplo em (i).
  3. [4 pontos]
    - (i) Suponha que inserimos, nesta ordem, as chaves `Q U E S T A O F C I L` em uma skip list inicialmente vazia. Suponha que o gerador de números aleatórios fornece os seguintes resultados, nessa ordem: `3 1 4 1 5 6 2 1 4 5 2` (assim, p.ex., a célula de Q fica com 3 ponteiros). Desenhe o resultado final dessas 11 inserções.
    - (ii) Descreva como funciona o processo de inserção da chave J na skip list que você obteve em (i); suponha que o gerador de números aleatórios especificou que esta chave deverá ser inserida em uma célula com 2 ponteiros. Em particular, diga quais são todas as comparações entre chaves que são executadas nesta inserção.

- (iii) Suponha que o prof. Nécio (PN) resolve usar uma skip list para manter um conjunto de inteiros positivos, usando a ordem usual dos inteiros para ordenar as chaves (e não a ordem alfabética). Suponha que PN decide usar um gerador de números aleatórios um tanto complicado, com o comportamento descrito a seguir. Para decidir quantos ponteiros usar em uma célula, o gerador examina o inteiro  $x$  presente na célula e faz o seguinte: se  $2^k$  divide  $x$  mas  $2^{k+1}$  não divide ( $k \in \mathbb{N}$ ), então o gerador devolve  $k + 1$ . Assim, p.ex., o inteiro 12 ficaria em uma célula com 3 ponteiros, enquanto que 7 ficaria em uma célula com 1 ponteiro.
- (a) Como é a skip list do prof. Nécio se as chaves presentes são os inteiros  $1, 2, \dots, 15$ ?
  - (b) Suponha agora que  $t \geq 1$  e o prof. Nécio insere os inteiros  $1, 2, \dots, 2^t - 1$  em uma skip list inicialmente vazia. Suponha agora que ele faz uma busca do inteiro  $2^t$ . Qual é a complexidade desta busca? (Estamos interessados no caso em que  $t$  é grande.) Você considera esta complexidade boa ou ruim?
  - (c) Suponha agora que o prof. Nécio remove todos os números pares de sua lista em (b). Como fica sua skip list? PN agora faz uma busca do inteiro  $2^t$ . Qual é a complexidade desta busca? Você considera esta complexidade boa ou ruim?
  - (d) Explique ao prof. Nécio por que é fácil para um usuário maldoso gerar instâncias ruins para o esquema de skip lists dele. Por que o esquema usual de skip lists não tem esse defeito?