

**MAC 338 – Análise de Algoritmos**  
PRIMEIRO SEMESTRE DE 2008  
Terceira Prova – 3 de julho

Nome do aluno: \_\_\_\_\_ Curso: \_\_\_\_\_

Assinatura: \_\_\_\_\_

No. USP: \_\_\_\_\_ Professor: \_\_\_\_\_

### Instruções

1. Não destaque as folhas deste caderno.
2. A prova pode ser feita a lápis.
3. A legibilidade também faz parte da nota!
4. A prova consta de 4 questões. Verifique antes de começar a prova se o seu caderno de questões está completo.
5. Não é permitido o uso de folhas avulsas para rascunho.
6. Não é necessário apagar rascunhos no caderno de questão mas especifique qual é a resposta e qual é o rascunho.
7. A prova é sem consulta.

**Não escrever nesta parte da folha**

Questão	Nota	Observação
1		
2		
3		
4		
Total		

**Boa prova!**

1. [3,0 pontos]

Projete uma estrutura de dados que suporte as seguintes duas operações sobre um conjunto  $S$  de inteiros:

(a) **insere**( $S, x$ ): insere  $x$  no conjunto  $S$ ;

(b) **remova\_maior\_metade**( $S$ ): remove os maiores  $\lceil |S|/2 \rceil$  elementos de  $S$ .

Explique como implementar essa estrutura de dados de maneira que  $m$  operações consumam tempo  $O(m)$ .

Mais precisamente, diga como  $S$  é armazenado, e escreva os dois algoritmos acima. Diga quanto tempo cada um deles consome no pior caso. Depois mostre que uma seqüência arbitrária de  $m$  chamadas a **insere** ou **remova\_maior\_metade** sobre um conjunto  $S$  inicialmente vazio tem custo total  $O(m)$  no pior caso.

2. [1,5 pontos]

Considere a implementação do union-find por árvores enraizadas. Segue a implementação recursiva do FINDSET com compressão de caminhos.

```
FINDSET ( $x$ )  
1 se  $x \neq \text{pai}[x]$   
2   então  $\text{pai}[x] \leftarrow \text{FINDSET}(x)$   
3 devolva  $\text{pai}[x]$ 
```

Escreva uma versão não recursiva do FINDSET com compressão de caminhos. O consumo de tempo da sua função deve ser assintoticamente igual ao da acima e o consumo de espaço extra deve ser  $O(1)$ .

3. [3,0 pontos]

Mostre que, se todos os caracteres do padrão  $P[1..m]$  são distintos, o algoritmo ingênuo que busca  $P$  em um texto  $T[1..n]$  pode ser modificado para consumir tempo  $O(n)$ .

Mais precisamente, escreva uma versão modificada do algoritmo ingênuo de busca de padrão que, dado  $T[1..n]$  e  $P[1..m]$  onde todos os caracteres de  $P$  são distintos, consome tempo  $O(n)$  para imprimir todas as posições do texto  $T[1..n]$  em que  $P[1..m]$  aparece. Mostre que seu algoritmo de fato consome tempo  $O(n)$  no pior caso.

4. [1,0 + 1,5 pontos]

Seja  $G = (V, E)$  um grafo. Um conjunto  $S \subseteq V$  é *independente* se quaisquer dois vértices de  $S$  não são adjacentes. Ou seja, não há nenhuma aresta do grafo com as duas pontas em  $S$ . O problema IS consiste no seguinte: dado um grafo  $G$  e um inteiro  $k \geq 0$ , existe um conjunto independente em  $G$  com  $k$  vértices? Mostre que IS é NP-completo.