

# MAC5710 - Estrutura de Dados e suas Aplicações

Primeiro semestre de 2009

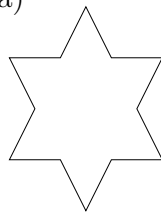
## Lista 2

1. Considere quatro vagões de trem numerados de 1 a 4, nesta ordem, na entrada de um estacionamento em forma de pilha. Liste todas as permutações de 1 a 4 e marque as que dão uma ordem dos vagões que pode ser obtida na saída deste estacionamento. Você vê algo em comum entre as permutações que não podem ser obtidas?
2. Passe a expressão aritmética abaixo para notação posfixa indicando, para cada caractere lido, o conteúdo da pilha de operadores.

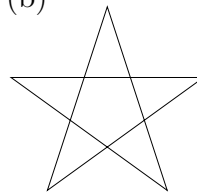
$$A * B - (C / D * E - F / G) + H$$

3. Dê o protótipo das sete operações que manipulam uma fila dupla com uma descrição de uma linha do que cada operação faz, como feito em aula para filas e pilhas.
4. Escreva um arquivo postscript que corresponda ao desenho de uma estrela de 6 pontas (figura abaixo, item (a)), usando a função `hill` vista em aula. Depois escreva um arquivo postscript com o desenho de uma estrela de 5 pontas (figura abaixo, item (b)).

(a)



(b)



5. No pior caso, quanto espaço (em notação assintótica) da pilha de execução pode consumir a implementação usual do quicksort para ordenar um vetor de tamanho  $n$ ? Como implementar o quicksort de modo a gastar (assintoticamente) menos que isso?
6. Considere o seguinte problema:

**SUBSETSUM:** Dado um número  $S$  e uma seqüência de  $n$  números, determinar todas as subseqüências da seqüência dada cuja soma é exatamente  $S$ .

Resolva o problema acima por força bruta usando o algoritmo de geração de subseqüências visto em aula.

7. Escreva um programa que leia uma seqüência de  $n$  (`(`, `[`, `{`, `}`, `]`, `)`) e, usando uma fila ou uma pilha (decida qual é mais conveniente), decide se a seqüência é bem formada.

