

## Exercícios referentes ao Capítulo 10: Estruturas Lineares

### 1. Exercícios de programação:

- (a) No TAD Fila que estudamos neste capítulo, os novos elementos sempre eram adicionados ao final da fila, esperando sua vez até serem removidos do começo da fila. Entretanto, em diversas situações na programação é útil estender nosso conceito de fila para uma versão de “**fila de prioridade**”, na qual a ordem dos elementos é determinada por um valor de prioridade numérica. Em uma fila de prioridade (*priority queue*), quando um item é enfileirado, ele deve ser colocado imediatamente à frente do item com uma prioridade menor do que a sua (e, por conseguinte, de todos os itens com prioridades menores). Se dois item tem a mesma prioridade, o item inserido antes deve ser removido antes também.

Sua tarefa neste exercício é modificar a interface `queueTAD.h` e a implementação com lista encadeada `queueTAD_lse.c` e adicionar uma nova função, com o seguinte protótipo:

```
queue_status priority_enqueue  
(queueTAD queue, elementoT elemento, int prioridade);
```

Os parâmetros “queue” e “elemento” são os mesmos da função `enqueue`; o parâmetro “prioridade” adicional é um inteiro que representa a prioridade, da seguinte maneira: inteiros menores correspondem a prioridades maiores, por exemplo: um elemento com prioridade 1 tem maior prioridade de que um elemento com prioridade 2.

Lembre-se de que você está implementando uma extensão à uma interface já existente e, portanto, você não pode “quebrar” a interface antiga. Clientes que não quiserem usar a extensão de inserir com prioridade não devem necessitar de fazer nenhuma alteração no código, ou seja: se um cliente já usar a interface `queueTAD.h`, ele não precisará fazer nenhuma alteração no código depois que você implementar a extensão para incluir a fila de prioridade. O cliente só precisará alterar o código se ele quiser passar a inserir elementos com prioridades.