

Capítulo 12

Estruturas lineares

1. Uma *fila de prioridades* é uma estrutura de dados composta por um certo número de filas, cada uma das quais associada a uma determinada prioridade.

Consideremos uma fila de prioridades com duas prioridades, urgente e normal. Nesta fila, novos elementos são adicionados, indicando a sua prioridade, e são colocados no fim da fila respetiva. Os elementos são removidos da fila através da remoção do elemento mais antigo da fila urgente. Se a fila urgente não tiver elementos, a operação de remoção remove o elemento mais antigo da fila normal. Existe uma operação para aumentar a prioridade, a qual remove o elemento mais antigo da fila normal e coloca-o como último elemento da fila urgente. As operações básicas para o tipo fila de prioridades (com prioridades urgente e normal) são as seguintes:

- *Construtores:*
 - $nova_fila_2p : \{\} \mapsto fila_2p$
 $nova_fila_2p()$ tem como valor uma fila de duas prioridades sem elementos.
- *Seletores:*
 - $inicio : fila_2p \mapsto elemento$
 $inicio(fila)$ tem como valor o elemento que se encontra no início da fila de prioridade *urgente* da *fila*; se a fila de prioridade *urgente* da *fila* não tiver elementos, tem como valor o elemento que se encontra no início da fila de prioridade *normal* da *fila*. Se as filas de prioridade *urgente* e *normal* não tiverem elementos, o valor desta operação é indefinido.
 - $comprimento_2p : fila_2p \times \{urgente, normal\} \mapsto \mathbb{N}_0$
 $comprimento_2p(fila, tipo)$ tem como valor o número de elementos da fila de prioridade *tipo* da *fila*.

- *Modificadores:*

- $coloca_2p : fila_2p \times \{urgente, normal\} \times elemento \mapsto fila_2p$
 $coloca_2p(fila, tipo, elm)$ altera de forma permanente a $fila$ para a fila que resulta em inserir elm no fim da fila de prioridade $tipo$ da $fila$. Devolve a fila resultante.
- $retira_2p : fila_2p \mapsto fila_2p$
 $retira_2p(fila)$ altera de forma permanente a $fila$ para: (1) a fila que resulta em remover o elemento que se encontra no início da fila de prioridade *urgente* da $fila$; (2) a fila que resulta em remover o elemento que se encontra no início da fila de prioridade *normal* da $fila$, se a fila de prioridade *urgente* da $fila$ não tiver elementos. Se as filas de prioridade *urgente* e *normal* não tiverem elementos, o valor desta operação é indefinido. Devolve a fila resultante.
- $aumenta_prioridade_2p : fila_2p \mapsto fila_2p$
 $aumenta_prioridade_2p(fila)$ altera de forma permanente a $fila$ para a fila que resulta em remover o elemento que se encontra no início da fila de prioridade *normal* da $fila$ e coloca-o no final da fila de prioridade *urgente* da $fila$. Se a fila de prioridade *normal* não tiver elementos, esta operação não altera a $fila$. Devolve a fila resultante.

- *Reconhecedores:*

- $e_fila_2p : Universal \mapsto logico$
 $e_fila_2p(arg)$ tem o valor *verdadeiro*, se arg é uma fila de prioridades com as prioridades *urgente* e *normal*, e tem o valor *falso*, em caso contrário.
- $fila_2p_vazia : fila \times \{urgente, normal\} \mapsto logico$
 $fila_2p_vazia(fila, tipo)$ tem o valor *verdadeiro*, se a fila de prioridade $tipo$ da $fila$ é a fila vazia, e tem o valor *falso*, em caso contrário.

- *Testes:*

- $filas_2p_iguais : fila_2p \times fila_2p \mapsto logico$
 $filas_2p_iguais(fila_1, fila_2)$ tem o valor *verdadeiro*, se $fila_1$ é igual a $fila_2$, e tem o valor *falso*, em caso contrário.

Defina a classe `fila_2_p` com prioridades **urgente** e **normal**.

2. O tipo de dados *lista circular* corresponde a uma lista na qual, excepto no caso de ser uma lista vazia, ao último elemento segue-se o primeiro. A Figura 12.1 (a) mostra esquematicamente uma lista circular em que o primeiro elemento é 4, o segundo, 3, o terceiro, 5, o quarto, 2 e o quinto (e último) é 1. Uma lista circular tem um elemento que se designa por primeiro elemento ou elemento do início da lista. Na lista da Figura 12.1 (a), esse elemento é 4. As listas circulares aparecem em várias aplicações,

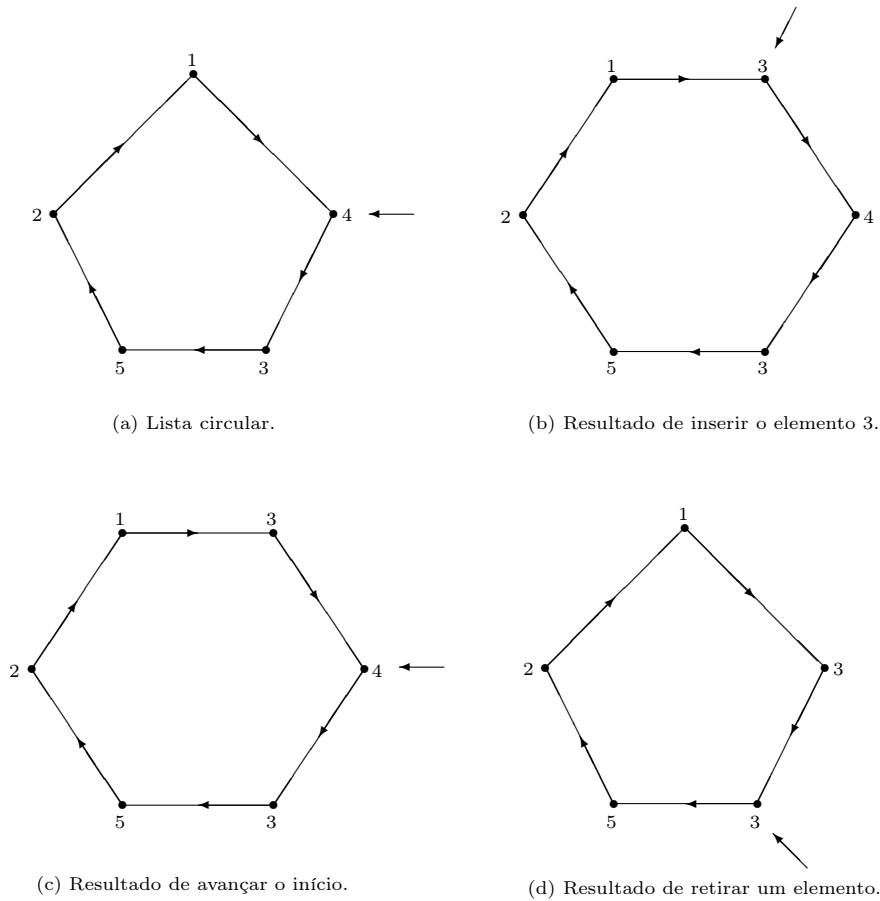


Figura 12.1: Operações sobre listas circulares.

sendo uma delas a representação de vídeos que se repetem ao chegar ao fim.

Com listas circulares, podemos efetuar as seguintes operações, as quais estão exemplificadas na Figura 12.1.

- *insere_circ*, que insere um elemento na lista. Com esta operação, o elemento inserido passa a ser o primeiro da lista resultante, o primeiro elemento da lista original passa a ser o segundo da nova lista, e assim sucessivamente.
- *primeiro_circ*, que inspeciona o primeiro elemento da lista, sem a alterar.
- *retira_circ*, que retira um elemento da lista. Com esta operação, o elemento retirado é sempre o do início da lista, passando o início

da lista resultante a ser o segundo elemento (se este existir) da lista original.

- *avanca_circ*, a qual avança o início da lista para o elemento seguinte. Esta operação não altera os elementos da lista, apenas altera o início da lista, que passa a ser o segundo elemento da lista original, se esta tiver pelo menos dois elementos; se apenas tiver um elemento, nada se altera; se a lista circular for vazia, esta operação tem um valor indefinido.
- (a) Especifique as operações básicas para listas circulares e classifique-as.
- (b) Supondo que a lista circular apresentada na alínea (a) da Figura 12.1 é representada externamente por @4, 3, 5, 2, 1@, defina a classe *lista_circular*. Permitindo a interação:

```
>>> lc = lista_circ()
>>> lc.insere_circ(1)
@1@
>>> lc.insere_circ(2)
@2, 1@
>>> lc.insere_circ(5)
@5, 2, 1@
>>> lc.insere_circ(3)
@3, 5, 2, 1@
>>> lc.insere_circ(4)
@4, 3, 5, 2, 1@
>>> lc.primeiro_circ()
4
>>> lc.el_n_circ(11)
3
>>> lc.insere_circ(3)
@3, 4, 3, 5, 2, 1@
>>> lc.avanca_circ()
@4, 3, 5, 2, 1, 3@
>>> lc.retira_circ()
@3, 5, 2, 1, 3@
```

3. Ao passo que uma pilha permite inserções e remoções numa das extremidades e uma fila permite inserções numa das extremidades e remoções na outra, uma *fila dupla*¹ é uma estrutura linear que permite inserções e remoções em ambas as extremidades.

- (a) Especifique as operações básicas para o tipo fila dupla.
- (b) Defina a classe *fila_dupla*.

¹Em inglês *double-ended queue* ou *deque*.