

Fila

Lista linear em que todas as inserções são feitas em uma das extremidades (fim) e todas as remoções são feitas na outra extremidade (início).

Implementação sequencial:

um vetor $F[1..MAX]$ e duas variáveis inteiras ini e fim .

Operações:

- Inicializa(F, ini, fim)
- Insere(F, ini, fim, x)
- Remove(F, ini, fim)
- Topo(F, ini, fim)
- Vazia(F, ini, fim)

Estruturas de Dados – p.9/11

Implementação das operações

Inicializa (F, ini, fim)

- 1 $ini \leftarrow 0$
- 2 $fim \leftarrow 0$

Insere (F, ini, fim, x)

- 1 $fim \leftarrow (fim \bmod MAX) + 1$
- 2 $F[fim] \leftarrow x$

Remove (F, ini, fim)

- 1 $ini \leftarrow (ini \bmod MAX) + 1$
- 2 **devolva** $F[ini]$

Primeiro (F, ini, fim)

- 1 **devolva** $F[(ini \bmod MAX) + 1]$

Vazia (F, ini, fim)

- 1 **se** $ini = fim$
- 2 **então devolva** VERDADE
- 3 **senão devolva** FALSO

Estruturas de Dados – p.10/11

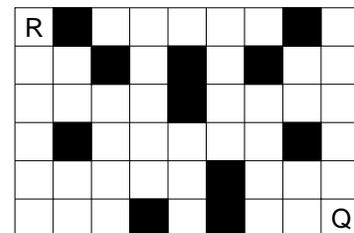
Aplicações

- cálculo de distância (Problema do ratinho)
- implementação de Round-Robin (escalonamento de processos numa CPU, RRDtool)

Round-Robin é uma técnica usada para distribuição de carga entre servidores ou para interpolar valores em um gráfico.

Estruturas de Dados – p.11/11

Problema do ratinho



Problema: Dado um labirinto e as posições do rato e do queijo no labirinto, determinar o número mínimo de passos que o rato precisa para chegar até o queijo.

Estruturas de Dados – p.12/11

Problema do ratinho

Entrada: inteiros m e n , matriz $A_{m \times n}$ tal que

$$A[i, j] = \begin{cases} -1 & \text{se } (i, j) \text{ é parede} \\ 0 & \text{se } (i, j) \text{ é livre} \end{cases}$$

e posições xr, yr do rato e xq, yq do queijo.

Saída: a distância do rato até o queijo.

Vamos assumir que a matriz A vem com uma moldura de -1 's para simplificar o algoritmo. Tal moldura, se ausente, pode ser acrescentada numa fase anterior.

Estruturas de Dados – p.13/11

Cálculo de distância

Distância (A, xr, yr, xq, yq)

- 1 **para todo** i e j **faça**
- 2 **se** $A[i, j] = 0$
- 3 **então** $D[i, j] \leftarrow \infty$
- 4 **senão** $D[i, j] \leftarrow -1$
- 5 $D[xr, yr] \leftarrow 0$
- 6 Inicializa(F, ini, fim)
- 7 Insere(F, ini, fim, xr, yr)

continua...

Estruturas de Dados – p.14/11

Cálculo de distância

```
8 enquanto não Vazia( $F$ ,  $ini$ ,  $fim$ ) faça
9    $x, y \leftarrow$  Remove( $F$ ,  $ini$ ,  $fim$ )
10  se  $D[x - 1, y] = \infty$ 
11    então  $D[x - 1, y] \leftarrow D[x, y] + 1$ 
12         Insere( $F$ ,  $ini$ ,  $fim$ ,  $x - 1, y$ )
13  se  $D[x + 1, y] = \infty$ 
14    então  $D[x + 1, y] \leftarrow D[x, y] + 1$ 
15         Insere( $F$ ,  $ini$ ,  $fim$ ,  $x - 1, y$ )
16  se  $D[x, y - 1] = \infty$ 
17    então  $D[x, y - 1] \leftarrow D[x, y] + 1$ 
18         Insere( $F$ ,  $ini$ ,  $fim$ ,  $x - 1, y$ )
19  se  $D[x, y + 1] = \infty$ 
20    então  $D[x, y + 1] \leftarrow D[x, y] + 1$ 
21         Insere( $F$ ,  $ini$ ,  $fim$ ,  $x - 1, y$ )
22 devolva  $D[xq, yq]$ 
```