



### *Commento all'es. 1*

La soluzione proposta utilizza un'interfaccia utente (un modulo di input modulare, nel quale la funzione `leggiComando` effettua la lettura di comandi, opzioni (dette “criterio”) ed eventuali altri dati (es. nome di file, nome o codice da cercare, ecc. Come risultato dell'input, viene preparato e restituito un dato composto (tipo `ComandoCompleto`), basato su due tipi `enum` (`Comando` e `Criterio`) e su una stringa.

Ad esempio, il comando: aggiorna nome “green” 6  
verrebbe ritornato come `{co_aggiorna, cr_nome, “green 6”}`

Si evitano più funzioni di ordinamento (per nome, codice, ecc) ricorrendo a una versione unificata (`ordinaCatalogo`), che riceve come parametro, oltre a vettore e dimensione, il criterio in base a cui ordinare (`crit`). Questo parametro viene semplicemente trasferito alla funzione unificata di confronto (`confrontaProdotti`).

La ricerca in base a un nome parziale pone il problema di poter trovare più di un risultato. Per questo si differenziano i casi di utilizzo dell'ordine (con ricerca dicotomica) oppure no, con ricerca lineare. La funzione `ricercaStampa`:

- nel caso di utilizzo della ricerca dicotomica, trova uno qualunque dei prodotti corrispondenti, dal quale (se trovato) reperisce gli altri mediante determinazione di un intervallo: se presenti, altri prodotti corrispondenti al criterio sono adiacenti, nel vettore ordinato, al dato trovato
- nel caso di ricerca lineare, itera più ricerche, attivando ogni volta una ricerca nella parte di vettore successivo all'ultimo prodotto trovato. Si assume ovviamente che la funzione di ricerca ritorni successo al primo dei dati corrispondenti al criterio.

### *Commento all'es. 2*

Le funzioni di allocazione della matrice `malloc2dR` e `malloc2dP` seguono il modello presentato in Puntatori e strutture dati a pag. 76. Esse sono estese per ricevere come parametro il nome del file da cui verranno letti i dati e per fornire come parametri passati per riferimento il numero di righe e di colonne.

Per determinare il numero di diagonali/antidiagonali basta generalizzare quanto visto nel problema delle 8 regine, dove la scacchiera era quadrata:

- il numero di diagonali/antidiagonali è  $\text{numd} = \text{nr} + \text{nc} - 1$ ;
- la lunghezza massima di diagonali/antidiagonali è il minimo tra il numero di righe e di colonne  $\text{maxd} = (\text{nr} > \text{nc}) ? \text{nc} : \text{nr}$ ; Si osservi l'uso di una forma compatta di istruzione condizionale: se la condizione  $(\text{nr} > \text{nc})$  è vera a `maxd` viene assegnato `nc` altrimenti viene assegnato `nr`
- le diagonali sono identificate da interi nell'intervallo  $0 \dots \text{numd} - 1$
- la cella della matrice di indice  $i, j$  appartiene all'antidiagonale di indice  $i+j$  e alla diagonale di indice  $i-j+\text{offset}$ . Si osservi che:
  - per coerenza con la figura nel testo del laboratorio sono scambiate diagonali e antidiagonal rispetto a quanto visto nel problema delle 8 regine



- la somma di un offset serve a garantire indici non negativi. Il valore di offset si calcola generalizzando a matrici non quadrate quanto visto nel problema delle 8 regine:  $\text{offset} = (nr > nc) ? nr-1 : nc-1;$
- nel calcolo dell'antidiagonale per ottenere l'output di figura il ciclo sulle colonne è esterno, quello sulle righe è interno
- la lunghezza delle diagonali/antidiagonali segue una regola comune: iniziando dalle celle in alto a DX e basso a SX per le diagonali (in alto a SX e basso a DX per le antidiagonali), essa cresce da 1 fino a  $\text{maxd}-1$  e, se il numero di diagonali è dispari, la diagonale principale ha lunghezza  $\text{maxd}$ .