

NOTE SULLO SVOLGIMENTO DELLA PROVA:

- PER SVOLGERE L'ELABORATO, APRIRE il Dev-C++ (dal Menù **Avvio** (o **Start**) nella barra degli strumenti in fondo allo schermo, selezionare Programmi e quindi Dev-C++);
- PRIMA DI INIZIARE LO SVOLGIMENTO DELL'ELABORATO, selezionare la voce **Identifica studente** nel menù **Strumenti** all'interno dell'ambiente Dev-C++ e inserire i dati richiesti; se tale voce non è presente, proseguire con il punto successivo (l'identificazione verrà eseguita manualmente dal docente);
- per svolgere l'elaborato, aprire il progetto *esaInf.dev* presente nel directory *c:\esame\esaInf* e scrivere le funzioni richieste nel file *compito.cpp*, già presente nel progetto;
- per una corretta stampa dell'elaborato bisogna mantenere il codice entro i margini imposti dall'ambiente Dev-C++ (linea verticale presente alla destra della pagina).

Scaffale

Uno scaffale contenente pile di libri può essere rappresentato usando le seguenti costanti e i seguenti tipi:

```
const int MAXS = 100;
const int MAXL = 5;
const int MAXRIP = 3;
const int MAXPOS = 2;
```

```
struct Data {
    int giorno;
    int mese;
    int anno;
};
```

```
struct Libro {
    char tit[MAXS];
    char aut[MAXS];
    int pag;
    Data pubbl;
};
```

```
struct PilaLibri {
    Libro stack[MAXL];
    int top;
};
```

```
typedef PilaLibri Scaffale[MAXRIP][MAXPOS];
```

La struttura `Data` serve a memorizzare giorno, mese e anno di una data. La struttura `Libro` contiene il titolo di un libro, l'autore, il numero di pagine e la data di pubblicazione. La struttura `PilaLibri` rappresenta una pila di libri di grandezza massima pari a `MAXL` (il campo `stack` contiene i dati dei libri, mentre il campo `top` indica il livello di riempimento della pila). Il tipo `Scaffale` è definito come una matrice di oggetti di tipo `PilaLibri`. Le righe della matrice corrispondono ai ripiani, mentre le colonne alla posizione. I ripiani e le

posizioni sono numerati a partire da zero. Inizialmente lo scaffale è vuoto e il campo `top` di tutti gli oggetti di tipo `PilaLibri` è pari a `-1`.

Scrivere il corpo delle seguenti funzioni C++.

1. **`bool inserisciLibro(Scaffale s, const Libro l, int rip)`** che inserisce, se possibile, il libro `l` nel ripiano `rip` dello scaffale `s`. Il libro deve essere inserito in cima alla prima pila di libri non piena, partendo dalla posizione zero. Se l'inserimento va a buon fine la funzione restituisce *true*. Se invece tutte le pile di libri del ripiano in questione sono piene, allora la funzione restituisce *false*.
2. **`bool cercaAutore(const Scaffale s, int rip, int pos, const char a[])`** che cerca l'autore `a` all'interno della pila di libri che si trova nel ripiano indicato da `rip` e nella posizione indicata da `pos`. Se almeno uno dei libri ha come autore `a`, la funzione restituisce *true*; altrimenti restituisce *false*.
3. **`bool piuVecchio(const Scaffale s, Libro* pl)`** che cerca il libro più vecchio presente nello scaffale (relativamente alla data di pubblicazione). Il libro in questione viene restituito al chiamante riempiendo l'oggetto puntato da `pl`. La funzione restituisce *true* se lo scaffale contiene almeno un libro, *false* altrimenti.
4. **`bool sposta(Scaffale s, int ripsrc, int possrc, int ripdst, int posdst)`** che sposta la pila di libri presente sul ripiano `ripsrc` e in posizione `possrc` in un altro posto. Il nuovo posto ha ripiano indicato da `ripdst` e posizione indicata da `posdst`. Affinché lo spostamento possa aver luogo, la pila di partenza non deve essere vuota, mentre la pila di destinazione deve essere vuota. Nel caso in cui lo spostamento avvenga con successo la funzione restituisce *true*, altrimenti restituisce *false*. I libri dopo lo spostamento devono avere, nella pila di destinazione, un ordine inverso rispetto a quello che avevano nella pila di partenza (quello che era in cima alla pila diventerà quello in fondo alla pila, e così via).
5. **`bool salvaOrd(const Scaffale s, const char f[])`** che salva nel file di nome `f` i dati di tutti i libri presenti ordinandoli per numero di pagina decrescente, come illustrato dall'esempio a seguire. La funzione restituisce *true* se il salvataggio va a buon fine, *false* altrimenti. Esempio:

```
titolo2, autore2, 6/4/1987, 564
titolo4, autore4, 8/8/2004, 200
titolo1, autore1, 7/4/2000, 123
titolo3, autore3, 12/2/1999, 54
```