

Una banca possiede *NUM* sportelli per il servizio ai clienti. Gli sportelli possono essere aperti o chiusi. Quando un cliente entra nella banca, se uno degli sportelli aperti è libero accede direttamente allo sportello; altrimenti si mette in coda. La banca può essere simulata con le strutture seguenti:

```
struct sportello
{
    bool stato;          //true -> sportello aperto,
                        //false -> sportello chiuso
    bool occupato;      //true -> sportello occupato,
                        //false -> sportello libero
};
struct cliente {char nome[LUN]; int istArr;}
struct coda
{
    int front, back; cliente queue[DIM];}
struct banca
{
    sportello s[NUM];
    coda c;
}
```

La struttura **coda** implementa la coda dei clienti. Di ogni cliente vengono memorizzati il nome (al massimo di **LUN** caratteri) e l'istante di arrivo (l'istante viene espresso come numero di minuti trascorsi dall'istante di apertura della banca). La coda ha dimensione massima **DIM**. Quando è creata un'istanza di tipo **banca**, tutti gli sportelli sono aperti e liberi.

Scrivere il corpo delle seguenti funzioni C++.

1. **bool arrivo(banca* b, const char nome[], int istArr)** che implementa l'arrivo del cliente il cui nome è passato come secondo argomento alla funzione; il terzo argomento indica l'istante di arrivo del cliente; se uno degli sportelli è libero, il cliente viene inviato direttamente allo sportello (lo sportello diventerà quindi occupato), altrimenti viene inserito in coda; la funzione restituisce **false** se la coda è piena; altrimenti restituisce **true**.
2. **int conta(const banca* b)** che restituisce il numero dei clienti presenti nella banca (considerando sia quelli allo sportello che quelli in coda);
3. **int attesaMedia(const banca* b, int ist)** che restituisce il tempo medio di attesa dei clienti che all'istante **ist** passato come secondo argomento alla funzione sono in coda; l'istante **ist** è espresso come numero di minuti trascorsi dall'istante di apertura della banca;
4. **void stampa(const banca* b)** che stampa in ordine alfabetico i nomi e gli istanti di arrivo dei clienti che si trovano in coda; si consiglia di creare un vettore di appoggio di strutture **cliente** dove memorizzare i clienti da ordinare;
5. **void libero(banca* b, int numSpor)** che rende libero lo sportello **numSpor** permettendo che venga servito il cliente che si trova da più tempo in coda; se la coda è vuota lo sportello rimane libero.

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;
- 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);
- SE L'ELABORATO È STATO COMPILATO SENZA ERRORI, PRIMA DELLA CONSEGNA, selezionare la voce **Consegna Compito** nel menù **Strumenti** all'interno dell'ambiente Dev-C++ e premere il tasto INVIO fino a quando non sparisce la finestra che è stata attivata.