

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).

Ambulatorio

In un ambulatorio ci sono due laboratori: uno per le analisi del sangue e l'altro per gli elettrocardiogrammi. Ogni nuovo paziente che arriva lascia le proprie generalità (nome, cognome, età e sesso) e viene inserito nella coda del laboratorio relativo al tipo di esame clinico che deve effettuare (si supponga che un paziente possa eseguire un solo tipo di esame). L'ambulatorio relativo agli elettrocardiogrammi può servire anche i pazienti in attesa di un esame del sangue a patto che non ci sia nessuno in attesa di un elettrocardiogramma.

Un sistema per la gestione dell'ambulatorio può essere realizzato attraverso le seguenti costanti e i seguenti tipi:

```
const int MAXS = 100;
const int MAXPAZ = 20;
```

```
enum Lab {SANGUE, ELETTRIO};
enum Sesso {MASCHIO, FEMMINA};
```

```
struct Paziente {
    char nome[MAXS];
    char cognome[MAXS];
    int eta;
    Sesso sex;
};
```

```
struct CodaPazienti {
    Paziente paz[MAXPAZ];
    int front, back;
};
```

```
struct Ambulatorio {
    CodaPazienti codaSangue;
    CodaPazienti codaElettro;
};
```

I tipi `Sesso` e `Lab` vengono usati per rappresentare rispettivamente il sesso di un paziente e il tipo di esame che deve effettuare. La struttura `Paziente` contiene nome, cognome, età e sesso di un paziente. Il tipo `CodaPazienti` viene utilizzato per rappresentare una coda di pazienti in attesa di essere serviti (al più MAXPAZ-1 pazienti per ogni coda). La struttura `Ambulatorio` contiene due code: una relativa ai pazienti in attesa di eseguire analisi del sangue (`codaSangue`) e una relativa ai pazienti in attesa di un elettrocardiogramma (`codaElettro`).

Scrivere il corpo delle seguenti funzioni C++.

1. **`bool inserisci(Ambulatorio* pa, Paziente p, Lab e)`** che, se possibile, inserisce il paziente `p` nell'ambulatorio puntato da `pa`. In particolare, il paziente viene inserito nella coda del laboratorio indicato dal parametro `e`. L'inserimento non è possibile se la coda in cui il paziente deve essere inserito è piena, e in tal caso viene restituito il valore *false* al chiamante. Nel caso in cui, invece, il paziente venga inserito in coda, la funzione restituisce il valore *true*.
2. **`bool desiste(Ambulatorio* pa, const char n[], const char c[])`** che elimina dall'ambulatorio puntato da `pa` il paziente avente nome `n` e cognome `c` (si supponga che non ci siano casi di omonimia). Il paziente che desiste può essere in una qualunque delle due code. La funzione compatta opportunamente l'array `paz` della coda in cui si trovava il paziente che va via. La funzione restituisce *true* se un paziente con il nome e il cognome specificati era presente nell'ambulatorio, *false* altrimenti.
3. **`float etaMedia(const Ambulatorio* pa, Lab e)`** che restituisce l'età media dei pazienti presenti nell'ambulatorio puntato da `pa` e che sono nella coda specificata da `e` (la funzione restituisce *0* se non ci sono pazienti che devono effettuare tale esame).
4. **`void stampaOrdinata(const Ambulatorio* pa)`** che stampa a video l'elenco di tutti i pazienti presenti nell'ambulatorio puntato da `pa`. L'elenco deve essere ordinato alfabeticamente per cognome e deve avere il formato indicato dal seguente esempio:

<i>cognome</i>	<i>nome</i>	<i>età</i>	<i>sesso</i>
Bianchi	Luigi	73	M
Gialli	Carolina	98	F
Neri	Francesca	88	F
Rossi	Mario	39	M
5. **`bool servizio(Ambulatorio* pa, Lab e)`** che rappresenta la disponibilità al servizio da parte del laboratorio specificato da `e`. Il paziente presente da più tempo nella coda associata a tale laboratorio viene rimosso dalla coda. Caso particolare: il laboratorio per gli elettrocardiogrammi può servire anche i pazienti in attesa di un esame del sangue a patto che non ci sia nessuno in attesa di un elettrocardiogramma. La funzione restituisce *true* se viene servito un paziente, *false* altrimenti.