

NOTE SULLO SVOLGIMENTO DELLA PROVA:

- Fare partire il Dev-C++ (dal Menù **Avvio** (o **Start**) nella barra degli strumenti in fondo allo schermo, selezionare Programmi e quindi Dev-C++);
- attraverso il menu **File->Open...** (o **apri** se il Dev-C++ è in italiano) aprire il progetto *APRIQUESTO.dev* presente nel directory *c:\esame\esaInf*;
- scrivere le funzioni richieste nel file *compito.cpp* già presente nel progetto;
- salvare spesso in modo da non perdere il lavoro nel caso in cui il PC abbia un malfunzionamento (per salvare premere il bottone a forma di dischetto del Dev-C++);
- 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).

Listino

Un programma per la gestione del listino prezzi degli articoli di un supermercato può essere realizzato usando le seguenti costanti e i seguenti tipi:

```
const int MAXS = 100;
const int MAXA = 200;

enum Categ {ALIMENTARI, CASALINGHI, GIOCATTOLI};

struct Articolo {
    int codice;
    char descr[MAXS];
    Categ categoria;
    float prezzo;
};

struct Listino {
    Articolo art[MAXA];
    int numArt;
};
```

Il tipo `Articolo` contiene le informazioni relative a un articolo di un supermercato: il codice univoco (un intero), una descrizione testuale dell'articolo (una stringa), la categoria dell'articolo (appartenente al tipo `Categ`), e il prezzo dell'articolo (un float). La struttura `Listino` memorizza i dati degli articoli presenti nel supermercato nell'array `art` (al più `MAXA` articoli) e il numero effettivo di articoli in `numArt`.

Scrivere il corpo delle seguenti funzioni C++.

1. **int caricaListino(Listino* pl, const char nomef[])** riempie il listino con i dati contenuti nel file di nome *nomef*. Nel file i dati degli articoli sono memorizzati su righe separate e il numero di righe non è noto a priori. In ogni riga ci sono nell'ordine: un intero che rappresenta il codice, un carattere che indica la categoria (A per ALIMENTARI, C per CASALINGHI, G per GIOCATTOLI), la descrizione dell'articolo come un'unica parola, un float che indica il prezzo. La funzione restituisce il numero di articoli inseriti nel listino.
2. **void scontrino(const Listino* pl, const int c[], const int q[], int n)** stampa a video uno scontrino. La funzione riceve in ingresso *n* codici e *n* quantità attraverso i due array *c* e *q*. L'elemento

i-esimo di q specifica quanti pezzi dell'articolo con il codice indicato dall'elemento i-esimo di c vengono acquistati. La funzione deve produrre una uscita a video analoga alla seguente:

```
1234 spaghetti_parilla x 2
Subtotale: 2.2
7654 caffe_lamazza x 1
Subtotale: 2.9
6666 supertele x 3
Subtotale: 15.6
-----
Totale: 20.7
```

3. **bool prezzoMax(const Listino* pl, const int c[], int n, Articolo* pa)** l'array c contiene n codici articolo. La funzione trova l'articolo con il prezzo più alto limitatamente a quelli con i codici specificati in c . L'articolo con il prezzo più alto viene restituito al chiamante riempiendo l'oggetto puntato da pa . La funzione restituisce *true* se il listino contiene almeno un articolo, *false* altrimenti.

4. **void stampaPrezziMedi(const Listino* pl)** la funzione stampa a video il prezzo medio degli articoli per tutte e tre le categorie, come illustrato dal seguente esempio:

```
Prezzi medi delle categorie Alimentari, Casalinghi, Giocattoli:
1.62667
6.125
9.56333
```

5. **int seleziona(const Listino* pl, char operatore, int cod, Articolo v[])** seleziona un sottoinsieme degli articoli presenti nel listino e li restituisce al chiamante riempiendo l'array v . Gli articoli sono selezionati secondo la seguente logica: il carattere `operatore` può valere `=`, `>`, o `<`; se `operatore` vale `=` viene selezionato l'articolo con codice uguale a `cod` (se presente); se `operatore` vale `>` vengono selezionati gli articoli con codice maggiore di `cod` (se presenti); se `operatore` vale `<` vengono selezionati gli articoli con codice minore di `cod` (se presenti). La funzione restituisce il numero di articoli inseriti in v .