

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).

Prodotti hitech

Un elenco di prodotti hitech può essere realizzato a partire dalle seguenti costanti e dai seguenti tipi:

```
const int MAXP = 100;
const int MAXN = 100;

enum Tipo {smartphone, pc, tablet, fotocamera};

struct Prodotto {
    Tipo tipo;
    char nome[MAXN];
    float prezzo;
    int gradimento;
    int codice;
};

struct Elenco {
    Prodotto prod[MAXP];
    int numProd;
};
```

`Prodotto` memorizza i dati di un prodotto: `tipo` indica il tipo di prodotto, `nome` indica il nome del prodotto, `prezzo` è il prezzo in euro, `gradimento` è un valore intero compreso tra 0 e 100, `codice` è un numero intero univoco che identifica il prodotto. La struttura `Elenco` memorizza i prodotti nell'array `prod`; il numero effettivo di prodotti presenti è indicato da `numProd`.

Realizzare le seguenti funzioni:

int carica(Elenco* pe, const char n[]) carica nell'elenco i dati dei prodotti contenuti nel file di nome `n`. Ogni prodotto è memorizzato su una riga separata del file; in ogni riga sono contenuti nell'ordine: un carattere che specifica il tipo, una parola che indica il nome, un reale che indica il prezzo, un intero che indica il gradimento, un intero che indica il codice. I caratteri usati per codificare il tipo del prodotto sono: 't' per tablet, 's' per smartphone, 'p' per pc, 'f' per fotocamera. Il numero di prodotti contenuti nel file non è noto a priori. La funzione restituisce il numero di prodotti caricati.

Esempio di file di ingresso:

```
t Appol_Aipad 650.5 88 1234
s Motoroba_G 700 91 8899
p Soni_Lap1 810 88 6543
s Appol_Aifon 640 89 7777
f Canox_F1 2200 97 7000
t Gugol_Nex 460 86 2211
```

```

s Zanzung_M1 450 78 6655
f Nikom_N1 2300 98 8123
s Zanzung_M2 510 81 6644
p Appol_book 1200 92 9876
f Nikom_N2 3400 96 5555

```

void stampaClassifica(const Elenco* pe) stampa a video il contenuto dell'elenco suddividendo i prodotti per tipo e, all'interno della stessa categoria, in ordine di gradimento decrescente. La stampa deve essere eseguita secondo quanto illustrato dal seguente esempio:

```

FOTOCAMERA:
Nikom_N1 2300 98 8123
Canox_F1 2200 97 7000
Nikom_N2 3400 96 5555

```

```

TABLET:
Appol_Aipad 650.5 88 1234
Gugol_Nex 460 86 2211

```

```

PC:
Appol_book 1200 92 9876
Soni_Lap1 810 88 6543

```

```

SMARTPHONE:
Motoroba_G 700 91 8899
Appol_Aifon 640 89 7777
Zanzung_M2 510 81 6644
Zanzung_M1 450 78 6655

```

int seleziona(const Elenco* pe, float pmin, float pmax, Tipo t, int v[]) inserisce nell'array `v` i codici dei prodotti aventi il tipo specificato da `t` e prezzo compreso tra `pmin` e `pmax`. La funzione restituisce il numero di valori inseriti in `v`.

int rimuovi(Elenco* pe, const char s[]) rimuove dall'elenco tutti i prodotti il cui nome contiene la sottostringa `s`. La funzione restituisce il numero di elementi rimossi.

void prezzoMinMax(const Elenco* pe, const int cods[], int n, float* pmin, float* pmax) la funzione riceve in ingresso `n` codici attraverso il parametro `cods` e calcola il prezzo minimo e massimo dei prodotti dell'insieme. Il prezzo minimo e massimo vengono restituiti al chiamante riempiendo gli oggetti puntati da `pmin` e `pmax` rispettivamente.