

## Contesto

Un file di testo memorizza le informazioni relative agli accessi ai 15 diversi impianti di risalita di un comprensorio sciistico. L'accesso agli impianti è consentito ai soli utenti muniti di skipass. Ogni skipass e ogni impianto hanno un proprio codice (max 6 caratteri). Esistono tre diverse tipologie di skipass: stagionale (S), giornaliero (G) e a punti (P). Ad ogni accesso a un impianto con skipass a punti verrà scalato un numero di punti specifico per ogni impianto. Il costo di un singolo punto è pari a 5 centesimi di euro. Ogni accesso viene registrato memorizzando le seguenti informazioni (separate da carattere di tabulazione):

- giorno dall'inizio della stagione sciistica
- codice impianto;
- punti da scalare;
- codice skipass;
- tipo di pass (S/G/P).

Il file è ordinato in ordine cronologico.

Un esempio del contenuto del file è il seguente:

```
1 I7 40 S254 G
1 I1 25 S2356 P
1 I12 50 S01A S
5 I2 10 S1 S
15 I1 25 S123 P
```

## Informazioni sul programma richiesto

Si scriva un programma in linguaggio C in grado di elaborare un file avente il formato descritto, al fine di restituire i risultati indicati nei punti specificati di seguito. Il programma deve poter essere invocato da linea di comando. Un esempio di invocazione è la seguente:

```
./a.out accessi.txt impianto
```

dove

- a.out è il nome del programma eseguibile da invocare;
- accessi.txt è il nome del file di dati da elaborare.
- impianto è il codice dell'impianto a cui si devono riferire le elaborazioni richieste al punto 2.

**IMPORTANTE:** il programma finale dovrà produrre la stampa di risultati esattamente col formato specificato nei vari punti. In particolare, *non aggiungere all'output del testo non richiesto*.

Eventuali righe di output aggiuntive che si vogliono generare in fase di debug, ma che si vogliono escludere dai test, possono essere stampate includendo in prima posizione il carattere #.

**NOTA:** Il buon funzionamento del programma può essere verificato col comando

```
./pvcheck ./a.out
```

dove a.out è il nome del file eseguibile.

# **RICHIESTE**

## **1 Numero di accessi**

Calcoli il numero totale di accessi riportati nel file.  
Stampare il numero con il seguente formato:

```
[TOTALE_ACCESSI]  
numero_accessi
```

## **2 Numero accessi per tipo di pass di un impianto**

Specificato il codice di un impianto (fornito in input come secondo parametro da linea di comando), calcolare e stampare il numero totale di accessi con pass di tipo G, P e S, con il seguente formato:

```
[ACCESSI_IMPIANTO]  
accessi_pass_G  
accessi_pass_P  
accessi_pass_S
```

## **3 Massimo numero accessi**

Determinare l'impianto (o gli impianti) che ha(nno) avuto il numero massimo di accessi.  
Stampare il risultato col seguente formato:

```
[IMPIANTO_MAX]  
codice_impianti_max_accessi1  
codice_impianti_max_accessi2  
...
```

## **4 Giornata di punta**

Si calcoli la giornata di punta, ossia quella che ha avuto il numero massimo di accessi. Se ci sono più giornate con il massimo numero di accessi, si stampi la prima della stagione.  
Stampare il risultato col seguente formato:

```
[GIORNO_PUNTA]  
primo_giorno_di_punta
```

## **5 Totale incasso punti**

Calcolare il totale degli incassi, limitatamente agli accessi con pass a punti. Si stampi tale valore con il seguente formato:

```
[INCASSO_PUNTI]  
totale_incasso_punti
```

...: CONTINUA SULL'ALTRO LATO ...

NOTA: stampare il valore con due cifre decimali usando lo specificatore di formato "%.21f".

## Esempio

Un esempio di output formattato correttamente è il seguente (i valori corrispondono ai dati di esempio in cui il punto 2 viene calcolato per l'impianto I1):

[TOTALE\_ACCESSI]

5

[ACCESSI\_IMPIANTO]

0

2

0

[IMPIANTO\_MAX]

I1

[GIORNO\_PUNTA]

1

[INCASSO\_PUNTI]

2.50

## Note

- Il programma deve essere composto da un singolo file sorgente salvato nella directory di lavoro;
- il nome del file deve coincidere con il cognome dello studente con l'aggiunta dell'estensione .c (esempio **rossi.c**)
- un programma che non può essere compilato viene automaticamente considerato insufficiente;
- la prima riga del programma deve essere un commento che riporta nome, cognome e numero di matricola;
- oltre alla correttezza, vengono valutati positivamente aspetti quali la leggibilità del programma, una buona formattazione del sorgente, l'uso appropriato dei commenti, generalità del codice e l'organizzazione del codice in funzioni;
- è possibile far uso di manuali o testi, ma non di eserciziari (raccolte di esercizi risolti)
- non è consentito connettere al calcolatore qualunque tipo di dispositivo personale (pendrive, dischi fissi, telefoni. . . );
- **tentativi di uso improprio di mezzi informatici e non informatici atti, ad esempio, alla comunicazione con studenti o altri soggetti saranno deferiti alla commissione disciplinare di Ateneo.**