

La cena degli allergici
Progetto di Programmazione I (turni I e serale)
Febbraio 2000

La contessa Sabidaroli-Bandrillussi deve organizzare una cena, a cui sono invitati un certo numero di suoi conoscenti. Ciascuno degli invitati ha un certo numero di intolleranze alimentari verso determinati ingredienti. Lo scopo è quello di verificare se un menù sia tale per cui, per ogni portata, ciascun invitato possa scegliere almeno un piatto che non contenga ingredienti cui egli è allergico.

Il problema viene specificato mediante tre file, come segue:

File degli invitati invitati.dat

Il file degli invitati (di nome `invitati.dat`) contiene, su ciascuna riga, il nome dell'invitato seguito dall'elenco degli ingredienti cui egli è allergico (si intende che nomi e ingredienti siano stringhe alfabetiche non contenenti spazi, di lunghezza minore o uguale a 30 caratteri, e che siano separate fra loro da spazi).

In altre parole:

```
<nome ospite 1> <ingr> <ingr> <ingr>  
<nome ospite 2> <ingr> <ingr>  
...  
<nome ospite n> <ingr> <ingr> ... <ingr>
```

Ad esempio:

```
Paolo uova mele  
Mauro fegato  
Giovanni farina olio pesce
```

File delle ricette ricette.dat

Questo file (di nome `ricette.dat`) contiene, per ogni riga, il nome di un piatto, seguito dagli ingredienti.

Cioè:

```
<piatto 1> <ingr> <ingr> ...
<piatto 2> <ingr> <ingr> <ingr>
...
<piatto m> <ingr> <ingr> ...
```

Ad esempio:

```
amatriciana pasta pancetta pomodoro peperoncino parmigiano
pastacarbonara pasta uova pancetta pepe
risotrevisana riso trevisana burro parmigiano
pescecartoccio pesce rosmarino olio sale
tortamele farina uova burro zucchero mele
sainthonore farina uova panna zucchero burro cioccolato cognac
```

Menù menu.dat

Questo file (di nome menu.dat) contiene tre righe: sulla prima riga ci sono i primi piatti, sulla seconda riga i secondi e sulla terza riga i dessert.

Ogni riga contiene i nomi dei piatti, separati da spazi:

```
<primo piatto 1> <primo piatto 2> <primo piatto 3> ...
<secondo piatto 1> <secondo piatto 2> ...
<dessert 1> <dessert 2> ...
```

Ad esempio:

```
pastacarbonara risotrevisana
pescecartoccio
tortamele sainthonore
```

Output

Il programma deve analizzare il menù, e garantire che ciascun invitato possa mangiare almeno uno dei piatti di ciascuna portata. Negli esempi sopra indicati, Paolo può mangiare **risotrevisana** (mentre non potrebbe mangiare **pastacarbonara**, essendo allergico alle uova), può mangiare **pescecartoccio**; purtroppo, Paolo non può mangiare nessun dessert, perché è allergico alle mele e alle uova. Come conseguenza, il menù specificato **non** deve essere accettato.

Se anche uno solo degli invitati non può mangiare nessuno dei piatti previsti per una certa portata, il programma deve emettere sullo standard output (cioè sullo schermo) la stringa **Non accettato**. In caso contrario, il programma dovrà invece stampare la stringa **Accettato**.

Osservazioni.

1. Si osservi che non si prevede a priori nessun limite massimo né per il numero di invitati, né per il numero di ingredienti possibili, né per il numero di piatti.
2. I nomi dei file utilizzati dal programma devono essere **rigorosamente** quelli specificati sopra: i file devono essere esplicitamente aperti dal programma tramite una reset.
3. Tutte le stringhe che compaiono (i nomi degli invitati, gli ingredienti e i nomi dei piatti) sono stringhe alfabetiche di lunghezza minore o uguale a 30).
4. È lecito assumere che i piatti che compaiono nel file `menu.dat` siano presenti nel file `ricette.dat`, e inoltre che il file `ricette.dat` non contenga duplicazioni (cioè, ciascun piatto viene specificato una sola volta).
5. Il programma **non** deve emettere stringhe di output diverse da quelle indicate e **non** deve richiedere alcun input da tastiera.
6. All'inizio del programma deve essere presente un commento che indichi nome, cognome e numero di matricola dell'autore.
7. Il programma deve essere debitamente commentato; in particolare, deve essere presente un commento in cui vengono giustificate le strutture dati utilizzate.

Modalità di consegna.

1. Il programma, realizzato in Pascal standard, deve essere inviato per posta elettronica (in formato sorgente) ai seguenti indirizzi:
 - per gli studenti il cui cognome inizia con le lettere da A a E (inclusa): `boldi@dsi.unimi.it`;
 - per gli altri studenti: `ferram@dsi.unimi.it`.

La mail deve avere come subject **Progetto esame Programmazione** e deve contenere il nome, cognome e numero di matricola dello studente; il programma deve essere possibilmente accluso come attachment.

2. Il programma va inviato almeno **tre giorni lavorativi** prima della data fissata per l'orale.