

# X-Files

## Programmazione I

### Progetto d'esame (Aprile 2000)

#### 1 L'intruso

Una brutta sorpresa aspetta Fox Mulder e Dana Scully quando arrivano nel loro ufficio: nottetempo un intruso è riuscito a penetrare nell'archivio degli X-Files, e le cartelle relative si trovano sparse per terra. Prima ancora di scoprire chi possa essere stato, i due agenti devono riuscire a individuare quali e quanti documenti siano stati rubati. Così, mentre Mulder si mette sulle tracce del responsabile, Scully si siede al computer e scrive un programma che la aiuti a individuare quali sono i documenti mancanti.

#### 2 L'organizzazione dell'archivio

Gli X-Files sono organizzati in cartelle. Ogni cartella è individuata da un codice, rappresentato da una sequenza di numeri, e contiene al suo interno un certo numero di sottocartelle. Così, ad esempio, la cartella 22.3.15 conterrà un certo numero di sottocartelle i cui codici saranno 22.3.15.1, 22.3.15.2, ...

#### 3 Input al programma

Il programma di Scully è scritto in modo da prendere in input i dati di tutte le cartelle che si trovano sparse sul pavimento. I dati vengono immessi da *tastiera*.

Ogni riga contiene il codice della cartella. Un tipico input potrebbe essere il seguente:

```
4.5.5
2
1.3
4.5.7
4.5.3
```

Naturalmente dall'input si possono individuare alcune cartelle che mancano con certezza. Nell'esempio sopra indicato, le cartelle mancanti sono sicuramente le seguenti:

1.1  
1.2  
3  
4.1  
4.2  
4.3  
4.4  
4.5.1  
4.5.2  
4.5.3  
4.5.4  
4.5.6

Infatti, se esistono cartelle in cui il numero più esterno è 1,2,4 deve anche esistere una cartella di numero esterno 3. D'altra parte, se esiste una cartella che inizia con 4.5 dovranno anche esistere cartelle 4.1, 4.2, 4.3, 4.4, e così via.

## 4 Output del programma

Il programma di Scully riporta in output, a *video*, l'elenco delle cartelle che mancano *sicuramente*. Tali cartelle vengono emesse in ordine lessicografico, cioè vengono emesse per prime le cartelle che hanno il primo numero più basso, e fra le cartelle con lo stesso primo numero, quelle che hanno il secondo numero più basso ecc.

**NOTA:** Si può assumere che l'input sia sintatticamente corretto: ciascuna riga (compresa l'ultima) è terminata da un fine-linea, e non contiene spazi; i numeri che compongono ciascun codice sono separati da un punto. L'input è terminato da un fine-file.

## 5 Modalità di consegna

1. Il programma, realizzato in Pascal standard, deve essere inviato per posta elettronica. Il programma deve contenere nella prima riga, inseriti come commento Pascal nome, cognome e numero di matricola dello studente.
2. Va inviato esclusivamente il file sorgente, che deve essere un semplice file `ascii` incluso come attachment alla mail.
3. La mail va indirizzata:
  - 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`.

Deve avere come subject

Programmazione <numero di matricola>

e deve contenere nel testo il nome, cognome e numero di matricola dello studente.

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

Infine si ricorda che il progetto va sviluppato individualmente.