

Laboratorio di Linguaggi di Sistema – a.a. 2008/2009 – Progetto finale III (base)

Il problema

Come parte di un sistema di servizi di supporto al turismo, dovete realizzare un modulo che offra delle funzioni per la pianificazione di viaggi intermodali basati sul trasporto pubblico. Il sistema deve consentire di specificare un punto di partenza, un punto di destinazione, e proporre uno o più tragitti, sfruttando tutti i mezzi disponibili (autobus urbani ed extraurbani, ferrovie, traghetti, metropolitane, aerolinee, ecc.) per effettuare il viaggio richiesto.

Implementazione

Il modulo sarà costituito da un programma in C, destinato a essere eseguito all'interno di un sistema complesso; quest'ultimo sarà dotato di interfacce diversificate (totem, web, cellulari, ecc.). Il sistema di interfaccia fornisce al modulo di pianificazione la richiesta di un tragitto da effettuare, sotto forma di una singola riga di testo per ogni richiesta (che il modulo riceve sul suo *stdin*) nel formato:

```
data (gg/mm/aaaa)|ora partenza (hh:mm)|origine|destinazione
```

e deve restituire sul suo *stdout* una serie di opzioni (si adottino criteri di ragionevolezza per decidere quante e quali opzioni offrire). Ogni opzione di viaggio avrà il formato

```
id_opzione|durata_totale
nome_servizio|origine|ora_partenza|destinazione|ora_arrivo
nome_servizio|origine|ora_partenza|destinazione|ora_arrivo
nome_servizio|origine|ora_partenza|destinazione|ora_arrivo
<riga bianca>
```

Ogni soluzione di viaggio deve essere tale da garantire che i cambi (differenza fra ora di arrivo di un mezzo e ora di partenza del mezzo successivo) siano di almeno 10 minuti.¹ La fine delle opzioni è indicata da una riga contenente soltanto la stringa “END” (che appare al posto del nome di una opzione).

Esempio

Input:

```
1/8/2009|06:00|Pisa CPT Pratale 1|London Victoria Station
```

Output:

```
Opzione 1|8:07
Pisa CPT LAM Verde|Pisa CPT Pratale 1|06:18|Pisa Stazione FS|6:31
FS Regionale 5512|Pisa Stazione FS|6:55|Pisa Aeroporto|7:02
Volo FR5312|Pisa Aeroporto|9:00|London Stansted|10:50
Bus Express L1|London Stansted|11:30|London Victoria Station|14:25
```

```
Opzione 2|7:02
Pisa CPT LAM Verde|Pisa CPT Pratale 1|06:18|Pisa Stazione FS|6:31
FS Regionale 4120|Pisa Stazione FS|7:05|Firenze SMN|8:01
Autotrasporti Pirra FI|Firenze SMN|8:30|Firenze Aeroporto|9:20
Volo EZ51|Firenze Aeroporto|9:40|London Gatwick|11:50
Gatwick Express|London Gatwick|12:18|London Victoria Station|13:20
```

```
[...]
END
```

Configurazione

Il modulo leggerà le informazioni sui servizi disponibili da un certo numero di file di configurazione, tutti con lo stesso formato, che legge in sequenza; i nomi di tali file vengono passati al modulo sulla riga di comando al momento del lancio da parte del sistema. Ogni file descrive una serie di servizi di trasporto, con il seguente formato:

```
nome_servizio|giorni di effettuazione
-|nome stazione|ora partenza
ora arrivo|nome_stazione|ora partenza
```

¹ Questo intervallo è irrealistico nel caso di aeroporti e traghetti, e dovrebbe essere configurato a seconda della tipologia di cambio, ma lo accettiamo come semplificazione.

```
ora arrivo|nome_stazione|ora partenza
...
ora arrivo|nome_stazione|ora partenza
ora arrivo|nome_stazione|-
```

in cui i capolinea di arrivo e destinazione non hanno, rispettivamente, un orario di arrivo e uno di partenza, mentre le (eventuali) fermate intermedie dispongono di entrambi. I vari servizi sono separati da una riga bianca. Alcuni file di esempio sono scaricabili dal sito del corso. I giorni di effettuazione del servizio sono espressi da una stringa di cifre, che rappresenta i giorni in cui il servizio viene effettuato, con 1=lunedì. Per esempio, la stringa “135” rappresenta un servizio effettuato solo il lunedì, mercoledì e venerdì.

Poiché il sistema è destinato a servire un numero assai elevato di richieste, è fondamentale che le richieste vengano servite in tempi brevi: si curi quindi in modo particolare l'efficienza dell'implementazione (sarà opportuno valutare in questo senso sia le strutture dati che gli algoritmi sviluppati).

Altre informazioni

Il modulo di pianificazione sarà eseguito da un processo distinto da quello principale, è quindi accettabile (benché sgradevole) che esso termini in casi eccezionali. In questo caso, il sistema di interfaccia provvederà ad abortire la richiesta corrente e a lanciare una nuova istanza del modulo. È opportuno, nei casi di terminazione in seguito a errore, che il modulo invii sul proprio *stderr* una stringa nel formato “ABORT codice descrizione_errore”, per esempio “ABORT 31 out of memory in solAlloc()”, che il sistema principale potrà eventualmente interpretare o inserire fra i propri log per successive analisi. E' altresì possibile inviare su *stderr* altri messaggi, non iniziati per “ABORT”, a propria discrezione. Naturalmente, nel caso di terminazioni frequenti il costo di rilettura dei file di configurazione (che si può assumere saranno di dimensioni molto grandi) diventa rilevante.

Suggerimento: la libreria standard del C dispone di funzioni che facilitano la gestione di ore, tempi e date; si suggerisce di investigarne l'uso laddove necessario, anziché implementarne proprie varianti.

Modalità di consegna

L'elaborato deve essere consegnato improrogabilmente entro il 21 settembre 2009 alle ore 17:00, e deve essere costituito da una stampa del codice sorgente, adeguatamente formattato e commentato, da una stampa con esempi di esecuzione, e da una breve relazione scritta (3-5 pagine) che descrive il progetto stesso. La versione a stampa può essere consegnata direttamente al docente, oppure depositata presso il centralino del Dipartimento di Informatica. Tutto il materiale deve essere anche inviato via email al docente (gervasi@di.unipi.it), in forma elettronica, entro il termine indicato.