Laboratorio di Algoritmi e Strutture Dati

Docenti: M. Goldwurm, S. Aguzzoli

Appello del 6 Settembre 2006

Progetto "Videosorveglianza"
Consegna entro il 25 Settembre 2006

Il problema

Una catena di supermercati deve proteggere i propri depositi di merce con adeguati sistemi di videosorveglianza.

Ogni *deposito* D è strutturato come di seguito descritto:

1. D è costituito da un insieme finito $\{S_1, S_2, \ldots, S_u\}$ di *stanze*.
2. Fra due stanze S_i e S_j di D può esistere o meno un *corridoio* rettilineo che le collega. Se un corridoio tra S_i e S_j esiste, allora è unico. Due stanze collegate da un corridoio si dicono *attigue*.
3. D possiede una unica stanza di *ingresso* S_1.
4. Ogni stanza S di D è collegata con l'ingresso S_1 di D tramite un *unico* cammino $T_0, T_1, \ldots, T_k$, in questo modo:
   (a) $T_0 = S_1$ e $T_k = S$.
   (b) Per ogni $i = 0, 1 \ldots, k-1$, le stanze T_i e T_{i+1} sono collegate da un corridoio.
   (c) Per ogni $i, j = 0, 1, \ldots, k$, se $i \neq j$ allora $T_i \neq T_j$.

   Se $T_0, T_1, \ldots, T_k$ è il cammino che collega l'ingresso di D a $S = T_k$ allora S è a *distanza* k dall'ingresso di D.

Ad ogni stanza S di D è associato un *nome* $n(D)$ che è una stringa finita di lunghezza arbitraria sull'alfabeto $\{a, b, \ldots, z\}$ delle lettere minuscole. Stanze distinte hanno nomi diversi. A ogni stanza S è inoltre associato un numero intero positivo che denota il valore $v(S)$ della merce depositata in S.

Una stanza può essere *sorvegliata direttamente* collocandovi una videocamera. Una stanza attigua a una sorvegliata direttamente risulta essere *sorvegliata indirettamente*. Un *impianto* di videosorveglianza è costituito dal posizionamento di videocamere nelle stanze in modo tale che ogni stanza sia sorvegliata direttamente o indirettamente. Per questioni economiche non è permesso collocare una videocamera in una stanza sorvegliata indirettamente. È invece possibile che una stanza sia sorvegliata indirettamente da più di una videocamera.

Sia I un impianto per un deposito D e siano $S_1, \ldots, S_k$ le stanze di D in cui è collocata una videocamera. Il *valore* $v(I)$ dell'impianto I è definito come segue:

$$v(I) \ = \ \sum_{i=1}^{k} v(S_i)$$

Un impianto I è *ottimale* per il deposito D se $v(I)$ ha valore massimo (ossia, per ogni impianto I' per D, si ha $v(I') \leq v(I)$). Si noti che in genere per un deposito possono progettarsi più impianti ottimali.

Un corridoio risulta *buio* in un impianto I se entrambe le stanze che collega non sono sorvegliate direttamente.

Esempio

Si supponga che il deposito D sia composto da 16 stanze $S_1,\ldots, S_{16}$, dove S_1 è l'ingresso, e che i corridoi nel deposito e i valori delle stanze siano quelli rappresentati nella Figura 1. Un esempio di impianto I_1 per D è ottenuto collocando le videocamere nelle stanze S_1, S_5, S_6, S_{10}, S_{11}, S_{12} e S_{13}; il valore di I_1 è 236. Tale impianto non è ottimale, in quanto è possibile costruire un impianto I di valore 237 ponendo le videocamere nelle stanze S_3, S_4, S_5, S_6, S_{10}, S_{11}, S_{12} e S_{13}. L'impianto I è l'unico impianto ottimale per D. In I ci sono quattro corridoi bui, ossia quelli che collegano S_1 con S_2, S_2 con S_7, S_7 con S_8, S_7 con S_9.

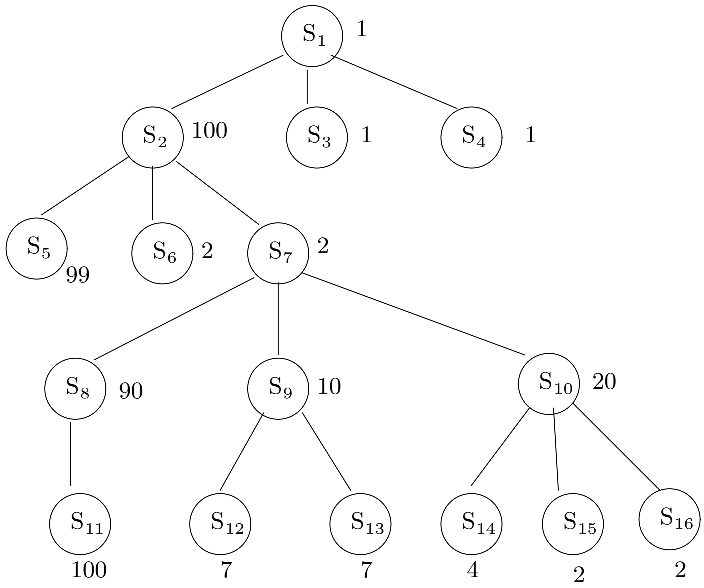

Figura 1: Esempio di deposito

Si richiede di implementare una struttura dati efficiente che permetta di eseguire le operazioni seguenti.

- **deposito** (α, v)

  Crea un deposito D specificandone la stanza di ingresso di nome α e valore v. Se esiste già una stanza di nome α in un deposito preesistente D', allora D' viene distrutto insieme a tutte le sue stanze e corridoi.

- **corridoio** (α, v, β)

  Se esiste già una stanza di nome α oppure non esiste ancora una stanza di nome β allora non viene compiuta alcuna operazione. Altrimenti introduce nel deposito D a cui appartiene β la stanza di nome α e valore v e un corridoio che collega α con β.

- **elimina** (α)

  Se non esiste una stanza di nome α non viene compiuta alcuna operazione. Se α è il nome dell'ingresso di un deposito D, allora elimina D con tutte le sue stanze e corridoi. Altrimenti ristruttura il deposito D che contiene la stanza di nome α come di seguito descritto: sia k la distanza della stanza S di nome α dall'ingresso di D, sia T l'unica stanza a distanza $k-1$ dall'ingresso di D che sia collegata ad α e sia $\{T_1, \ldots, T_h\}$ l'insieme di tutte le altre stanze collegate a S. Allora la stanza S e tutti i corridoi ad essa collegati vengono eliminati, mentre viene introdotto un corridoio che collega T a T_i per ogni $i = 1, \ldots, h$.

- **impianto** (α)

  Progetta un impianto ottimale per il deposito contenente la stanza di nome α e lo visualizza secondo il formato specificato nell'apposita sezione. Se non esiste alcuna stanza di nome α allora non viene eseguita alcuna operazione.

- **bui** (α)

  Progetta un impianto ottimale per il deposito contenente la stanza di nome α e visualizza l'elenco degli eventuali corridoi bui secondo il formato specificato nell'apposita sezione. Se non esiste alcuna stanza di nome α allora non viene eseguita alcuna operazione.

Si noti che le operazioni richieste sono liberamente implementabili; in particolare, non vanno necessariamente intese come prototipi di funzioni.

Specifiche di implementazione

Il programma deve leggere dallo standard input (`stdin`) una sequenza di righe (separate da `\n`), ciascuna delle quali corrisponde a una riga della prima colonna della Tabella 1, dove α e β sono stringhe finite sull'alfabeto $\{a, b, \ldots, z\}$ e v è un intero positivo. I vari elementi sulla riga sono separati da uno o più spazi. Quando una riga è letta, viene eseguita l'operazione associata; le operazioni di stampa sono effettuate sullo standard output (`stdout`), e ogni operazione deve iniziare su una nuova riga.

Riga di input	Operazione
`d` α v	**deposito** (α, v)
`c` α v β	**corridoio** (α, v, β)
`e` α	**elimina** (α)
`i` α	**impianto** (α)
`b` α	**bui** (α)
`f`	Termina l'esecuzione del programma

Tabella 1: Specifiche del programma

Note

1. Non devono essere presenti vincoli sul numero di depositi, di stanze e di corridoi (se non quelli determinati dal tipo di dato intero). Non si richiede – anzi si sconsiglia – l'uso di grafica, se non per test personali: in modo particolare, non si usi `conio.h` e neppure `clrscr()`.

2. Per semplicità si suppone che l'input sia sempre conforme alle specifiche di Tabella 1, per cui non è necessario controllare la correttezza dell'input. Per leggere l'input si usino le funzioni standard ANSI C `getchar()` e/o `scanf()`.

3. Sia $\{S_1, \ldots, S_n\}$ l'insieme delle stanze sorvegliate direttamente nell'impianto ottimale I calcolato dal comando `i` α. Sia v il valore di I e, per ogni $i = 1, \ldots, n$, sia $\alpha_i = n(S_i)$ il nome di S_i. Allora l'output del comando `i` α va visualizzato come segue:
   (v
   α_1
   α_2
   $\vdots$
   α_n
)
   L'ordine in cui le stanze sono elencate è irrilevante.

4. Sia $\{C_1, \ldots, C_n\}$ l'insieme dei corridoi bui nell'impianto ottimale calcolato dal comando `b` α. Per ogni $i = 1, \ldots, n$ siano S_i e T_i le due stanze collegate dal corridoio C_i e siano $\alpha_i = n(S_i)$ e $\beta_i = n(T_i)$ i nomi delle due stanze. Allora l'output del comando `i` α va visualizzato come segue:
   (
   α_1, β_1
   α_2, β_2
   $\vdots$
   α_n, β_n
)
   L'ordine in cui i corridoi sono elencati è irrilevante.

Esempio
Si supponga che le righe di input siano:

```
d frutta 4
d mobili 10
c ananas  6 frutta
c uva 13 frutta
c meloni 2  frutta
c pere 3 ananas
c mele 4  ananas
c banane 3 uva
c arance 3 uva
c noci 3 uva
c albicocche 10 meloni
c kiwi 5 meloni
c ciliegie 1 albicocche
c pesche 8 albicocche
c fragole 3 kiwi
c mandarini 3 kiwi
i noci
b meloni
e uva
e meloni
```

```
e albicocche
i ananas
b frutta
c armadi 10 mobili
c letti 1 armadi
c sedie 2 letti
i mobili
b letti
c tavoli 1 sedie
c cassapanche 1 sedie
i letti
c sgabelli 1 letti
c lettiacastello 1 letti
c armadietti 1 sedie
i mobili
f
```

L'output prodotto dal programma deve essere:

```
(37
uva
meloni
pere
mele
ciliegie
pesche
fragole
mandarini
)
(
frutta,ananas
)
(31
pere
mele
banane
arance
noci
ciliegie
pesche
fragole
mandarini
)
(
frutta,ananas
frutta,kiwi
)
(12
armadi
sedie
```

```
)
(
)
(13
mobili
letti
tavoli
cassapanche
)
(15
armadi
sgabelli
lettiacastello
tavoli
cassapanche
armadietti
)
```

Presentazione del progetto

Il progetto deve essere inviato per posta elettronica all'indirizzo `aguzzoli@dsi.unimi.it` entro il 25 Settembre 2006 (incluso). La discussione del progetto e l'esame orale si svolgeranno in data e luogo da specificarsi (consultare al riguardo il sito: `http://homes.dsi.unimi.it/~goldwurm/algo`).

Occorre presentare:

1. il codice sorgente (rigorosamente ANSI C, compilabile con **gcc**);
2. una sintetica relazione (formato pdf o rtf) che illustra le strutture dati utilizzate e analizza il costo delle diverse operazioni richieste dalla specifica.

I due o più file (file sorgenti C + relazione) devono essere contenuti in un unico file `.zip` il cui nome dovrà essere `cognome.zip`. La relazione e il codice devono riportare il vostro nome, cognome e matricola.

Una copia cartacea della relazione e del codice deve inoltre essere consegnata al dr. Aguzzoli entro il 25 Settembre 2006 (lasciandola eventualmente nella sua casella postale presso il dipartimento in via Comelico).

Si ricorda infine di presentarsi alla prova orale con una copia stampata della relazione e del codice.

Per ogni ulteriore chiarimento:
E-mail: `aguzzoli@dsi.unimi.it`
Ricevimento: il mercoledì, ore 15-16, stanza S204.

Avvisi

La versione aggiornata del progetto è pubblicata in .pdf sul sito:
`http://homes.dsi.unimi.it/~aguzzoli/algo.htm`.
Si consiglia di consultare periodicamente questo sito per eventuali correzioni e/o precisazioni relative al testo del progetto.

Si richiede allo studente di effettuare un adeguato collaudo del proprio progetto su numerosi esempi diversi per verificarne la correttezza e valutarne le prestazioni.

La realizzazione del progetto è una prova d'esame da svolgersi **individualmente**. I progetti giudicati frutto di **collaborazioni** saranno **estromessi** d'ufficio dalla valutazione.