Laboratorio di Algoritmi e Strutture Dati

Docenti: M. Goldwurm, S. Aguzzoli

Appello del 10 Gennaio 2006

Progetto "Mappe"
Consegna entro il 29 gennaio 2006

Il problema

Obiettivo del progetto è lo studio di mappe che specificano percorsi tra punti del piano.

Il *piano* è definito dall'insieme dei punti (x, y) tali che x e y sono numeri interi; ad ogni punto P è associato un valore, denotato con $Val(P)$.

Una *mappa* M è descritta da nome, origine e specifica.

- Il *nome* di M è una stringa di lunghezza arbitraria costruita sull'alfabeto $\mathcal{A}$ dei caratteri alfanumerici.
- L'*origine* di M è un punto del piano che denota il punto di partenza del percorso descritto dalla mappa.
- La *specifica* di M è una stringa finita di lunghezza arbiraria costruita sull'alfabeto $\{N, E, S, W\}$, che permette di determinare i punti costituenti il percorso.

Scriviamo $M = \langle w, P, \alpha \rangle$ per specificare che la mappa M ha nome w, origine P e specifica α. Supponiamo che la specifica α di M sia la stringa $s_1 s_2 \ldots s_n$; il *percorso* di M, denotato con $\pi(M)$, è la sequenza di punti $(P_0, \ldots, P_n)$ tali che:

- P_0 coincide con l'origine P di M.
- Sia $P_k = (x_k, y_k)$ il k-esimo punto della sequenza (con $0 \leq k < n$). Allora:

  $P_{k+1} = (x_k, y_k + 1)$ se $s_{k+1} = N$ (spostamento verso Nord);
  $P_{k+1} = (x_k + 1, y_k)$ se $s_{k+1} = E$ (spostamento verso Est);
  $P_{k+1} = (x_k, y_k - 1)$ se $s_{k+1} = S$ (spostamento verso Sud);
  $P_{k+1} = (x_k - 1, y_k)$ se $s_{k+1} = W$ (spostamento verso Ovest).

Chiamiamo P_n la *destinazione* della mappa M. Si noti che, dato un percorso $\sigma = (P_0, \ldots, P_n)$, la specifica α che produce la sequenza σ a partire dall'origine P_0 è univocamente determinata.

Data una stringa β finita sull'alfabeto $\{N, E, S, W\}$, diciamo che una mappa $M = \langle w, P, \alpha \rangle$ ha *direzione* β se β è un prefisso di α (ossia, $\alpha = \beta \cdot \gamma$, dove la stringa γ può essere vuota).

Il *valore* $Val(M)$ di una mappa M è dato dal valore della destinazione di M.

La *riduzione* di una mappa consiste nell'eliminare dal suo percorso i cicli. Un *ciclo semplice* di origine P è una sequenza di punti $\delta = (P, P_1 \ldots, P_m, P_{m+1})$ tale che $P = P_{m+1}$ e, per ogni $1 \leq k \leq m$, $P \neq P_k$. L'operazione di riduzione di una sequenza di punti $\sigma = (P_0, \ldots, P_n)$ avviene eliminando tutti i cicli semplici in σ secondo la seguente procedura:

- Sia P il punto di indice minimo tale che P è l'origine di un ciclo semplice δ; la sequenza σ può essere scritta come la concatenazione delle sequenze $\sigma_1 \cdot \delta \cdot \sigma_2$, dove σ_1 e σ_2 sono univocamente determinate e possono essere vuote. L'eliminazione di δ da σ produce la sequenza $\sigma' = \sigma_1 \cdot P \cdot \sigma_2$.

La procedura va riapplicata fino alla eliminazione di tutti i cicli semplici; si noti che la sequenza finale è univocamente determinata.

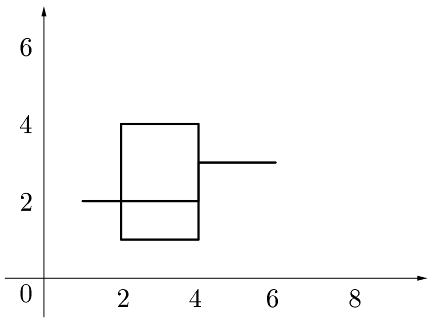
La riduzione di una mappa $M = \langle w, P, \alpha \rangle$ è data dalla mappa $M_r = \langle w, P, \alpha_r \rangle$ avente stesso nome e origine di M e tale che $\pi(M_r)$ è la riduzione di $\pi(M)$.

Si osservi che la riduzione M_r di una mappa M può produrre una specifica descritta dalla stringa vuota. In tal caso $\pi(M_r)$ è costituito da un solo punto.

Esempio

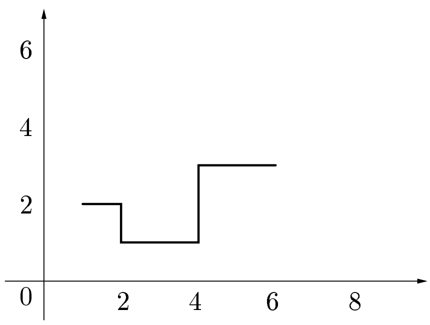
Sia $M = \langle mappa1,\ (1,2),\ EEENNWWSSSEENNEE \rangle$ una mappa. Il percorso $\pi(M)$ è la sequenza di punti

$$(1,2),(2,2),(3,2),(4,2),(4,3),(4,4),(3,4),(2,4),(2,3),(2,2),(2,1),(3,1),(4,1),(4,2),(4,3),(5,3),(6,3)\,.$$



Nel percorso $\pi(M)$ sono presenti tre cicli semplici, la cui origine è nei punti $P_1 = (2,2)$, $P_3 = (4,2)$, $P_4 = (4,3)$. Secondo le specifiche del processo di riduzione, il ciclo da eliminare è quello la cui origine è in P_1. L'eliminazione di questo ciclo produce la sequenza:

$$(1,2),(2,2),(2,1),(3,1),(4,1),(4,2),(4,3),(5,3),(6,3)\,.$$



Quest'ultima sequenza non contiene cicli, per cui il processo di riduzione termina. La riduzione M_r di M è la mappa $M_r = \langle mappa1,\ (1,2),\ ESEENNEE \rangle$.

Si richiede di implementare una struttura dati efficiente che permetta di eseguire le operazioni seguenti (si tenga presente che la minima porzione rettangolare di piano contenente tutti i percorsi specificati da mappe può essere molto grande rispetto al numero di mappe presenti nel piano, quindi *non è sicuramente efficiente rappresentare il piano mediante una matrice*).

- **crea** (n)

  Crea un nuovo piano, eliminando tutte le mappe e ponendo il valore di tutti i punti uguale a n.

- **valore** (x, y, n)

  Pone il valore del punto (x, y) uguale a n.

- **mappa** (w, x, y, α)

  Introduce la mappa $\langle w, (x, y), \alpha \rangle$. Se esiste una mappa M' di nome w, M' viene eliminata.

- **direzione** (β)

  Stampa i nomi di tutte le mappe di direzione β. Ogni nome va stampato su una nuova riga. L'ordine in cui i nomi vengono stampati non è rilevante.

- **riduci** (w)

  Sostituisce la mappa di nome w con la sua riduzione. Se non esiste alcuna mappa di nome w non esegue alcuna operazione.

- **usa** (β)

  Considera tutte le mappe di direzione β e ne elimina una qualunque fra quelle di valore massimo, ponendo il valore della sua destinazione uguale a 0.

- **stampa** (w)

  Stampa su una nuova linea l'origine e la specifica della mappa di nome w secondo il formato specificato nella prossima sezione. Se non esiste alcuna mappa di nome w non esegue alcuna operazione.

- **lista** ()

  Stampa tutti le mappe esistenti secondo il formato specificato nella prossima sezione.

Specifiche di implementazione

Il programma deve leggere dallo standard input (`stdin`) una sequenza di linee (separate da `\n`), ciascuna delle quali corrisponde a una linea della prima colonna della Tabella 1, dove x, y e n sono interi, w è una stringa non vuota sull'alfabeto $\mathcal{A}$ e α è una stringa non vuota sull'alfabeto $\{N, E, S, W\}$.

I vari elementi sulla linea sono separati da uno o più spazi. Quando una linea 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 linea.

Note

1. Non devono essere presenti vincoli sulla dimensione del piano, sul numero di percorsi e mappe, sulle lunghezze dei percorsi (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()`.

LINEA DI INPUT	OPERAZIONE
`c` n	**crea** (n)
`v` $x\ y\ n$	**valore** (x, y, n)
`m` $w\ x\ y\ \alpha$	**mappa** (w, x, y, α)
`d` α	**direzione** (α)
`r` w	**riduci** (w)
`u` α	**usa** (α)
`s` w	**stampa** (w)
`l`	**lista** ()
`f`	Termina l'esecuzione del programma

Tabella 1: Specifiche del programma

3. Sia $\langle w, P, \alpha\rangle$ una mappa, dove $P = (x, y)$ e $\alpha = s_1 \ldots s_k$. L'output di `s` w va visualizzato nel seguente formato:

```
w x y s1...sk
```

4. Siano $M_1 = \langle w_1, P_1, \alpha_1\rangle, \ldots, M_n = \langle w_n, P_n, \alpha_n\rangle$ tutte le mappe esistenti nel piano. L'output del comando `l` è

```
(
M1
...
Mn
)
```

dove ciascuna mappa va stampata secondo il formato del punto precedente. L'ordine in cui le mappe vengono elencate non è rilevante.

Esempio
Si supponga che le linee di input siano:

```
c 10
v 2 3 25
v 3 4 20
v 4 5 40
m prima 1 0 NWWSEENESSWNNNEN
m seconda 2 1 NNEEEEESSWWNNNEEESSWWWWNN
l
```

```
m terza 1 2 NNEESSWW
m quarta 5 5 W
l
d NNE
u NNE
m seconda2 2 1 NNEEEESSWWNNNNEEESSWWWWNN
r prima
r seconda2
l
d NNE
u NNE
l
s terza
r terza
s terza
l
c 100
m mappa1 -10 2 EEENNWWSEENWWSEEEE
m mappa2 -2 2 EEEENWSSEENNWWWWSW
m mappa3 3 5 EENEEENNWWSSSEEESSWWNNNEE
m mappa4 3 -4 ENNEESSWWSSESWNENNWN
d W
d EEEE
d EEE
v -5 3 50
v 9 6 70
v 4 -3 90
r mappa1
s mappa1
r mappa2
s mappa2
r mappa3
s mappa3
r mappa4
s mappa4
u E
l
f
```

L'output prodotto dal programma deve essere:

```
(
prima 1 0 NWWSEENESSWNNNEN
seconda 2 1 NNEEEESSWWNNNNEEESSWWWWNN
)
(
prima 1 0 NWWSEENESSWNNNEN
seconda 2 1 NNEEEESSWWNNNNEEESSWWWWNN
terza 1 2 NNEESSWW
quarta 5 5 W
```

```
)
seconda
terza
(
prima 1 0 NNEN
terza 1 2 NNEESSWW
quarta 5 5 W
seconda2 2 1 NNEN
)
prima
terza
seconda2
(
terza 1 2 NNEESSWW
quarta 5 5 W
seconda2 2 1 NNEN
)
terza 1 2 NNEESSWW
terza 1 2
(
terza 1 2
quarta 5 5 W
seconda2 2 1 NNEN
)
mappa2
mappa1
mappa2
mappa1 -10 2 EEENEE
mappa2 -2 2
mappa3 3 5 EENESENEE
mappa4 3 -4 EN
(
mappa1 -10 2 EEENEE
mappa2 -2 2
mappa3 3 5 EENESENEE
)
```

Presentazione del progetto

Il progetto deve essere inviato per posta elettronica all'indirizzo `aguzzoli@dsi.unimi.it` entro il 29 Gennaio 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 31 Gennaio 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.