Laboratorio di Algoritmi e Strutture Dati

Docenti: M. Goldwurm, S. Aguzzoli

Appello del 4 Febbraio 2003

Progetto "Itinerari"
Consegna entro il 28 Febbraio 2003

Il problema

Obiettivo del progetto è lo studio di itinerari su strade costruite nel piano. Una strada è specificata da una coppia di interi (x, y) che denota l'*origine* della strada e da una sequenza finita α costruita sull'alfabeto $\{N, E, S, W\}$.

Chiamiamo *tratto* un segmento di lunghezza unitaria disposto orizzontalmente o verticalmente nel piano. Formalmente, un tratto è un insieme di punti della forma

$$\{(k, y) : x \leq k \leq x + 1, \quad x, y \in \mathbb{Z},\ k \in \mathbb{R}\}$$

o della forma

$$\{(x, k) : y \leq k \leq y + 1, \quad x, y \in \mathbb{Z},\ k \in \mathbb{R}\}$$

Un tratto è denotato mettendo fra parentesi quadre i punti estremi del segmento in un ordine qualsiasi. Ad esempio, $[(2, 5), (3, 5)]$ denota il tratto $\{(k, 5) : 2 \leq k \leq 3\}$ (segmento di estremi $(2, 5)$ e $(3, 5)$); questo stesso tratto può essere denotato con $[(3, 5), (2, 5)]$.

La strada $\sigma((x, y), \alpha)$ descritta dalla sequenza α e di origine (x, y) è una sequenza di tratti

$$[(x_0, y_0), (x_1, y_1)], [(x_1, y_1), (x_2, y_2)], \ldots, [(x_{n-1}, y_{n-1}), (x_n, y_n)]$$

costruita come segue:

- Il punto (x_0, y_0) *origine* della strada è (x, y).
- Ogni simbolo di α corrisponde all'aggiunta alla strada di un tratto nella direzione specificata dal simbolo; in particolare, l'$(i + 1)$-esimo tratto $(0 \leq i \leq n - 1)$ sarà così costruito:
  - (Spostamento verso Nord)
    Se l'$(i + 1)$-esimo simbolo è N allora l'$(i + 1)$-esimo tratto sarà $[(x_i, y_i), (x_i, y_i + 1)]$.
  - (Spostamento verso Est)
    Se l'$(i + 1)$-esimo simbolo è E allora l'$(i + 1)$-esimo tratto sarà $[(x_i, y_i), (x_i + 1, y_i)]$.
  - (Spostamento verso Sud)
    Se l'$(i + 1)$-esimo simbolo è S allora l'$(i + 1)$-esimo tratto sarà $[(x_i, y_i), (x_i, y_i - 1)]$.
  - (Spostamento verso Ovest)
    Se l'$(i + 1)$-esimo simbolo è W allora l'$(i + 1)$-esimo tratto sarà $[(x_i, y_i), (x_i - 1, y_i)]$.

Nel seguito si considerano *solamente* strade in cui non ci sono cicli (ossia, non vi è punto del piano per cui una stessa strada passi più di una volta); è possibile invece che un tratto o un punto del piano appartenga a più di una strada.

Dati due punti (a,b) e (c,d) del piano, un *itinerario* da (a,b) a (c,d) è una sequenza di tratti

$$[(x_0,y_0),(x_1,y_1)],[(x_1,y_1),(x_2,y_2)],\ldots,[(x_{k-1},y_{k-1}),(x_k,y_k)] \qquad (*)$$

tale che:

1 (x_0,y_0) coincide con (a,b);

2 per ogni $1 \le i \le k$, il tratto $[(x_{i-1},y_{i-1}),(x_i,y_i)]$ appartiene ad almeno una strada del piano;

3 ogni punto del piano non vi compare più di due volte;

4 (x_k,y_k) coincide con (c,d).

Nel costruire un itinerario, un tratto di una strada può essere percorso in entrambi i sensi. Per semplicità d'ora in poi denoteremo l'itinerario $(*)$ con

$$(x_0,y_0),(x_1,y_1),(x_2,y_2),\ldots,(x_{k-1},y_{k-1}),(x_k,y_k)$$

mettendo in evidenza i punti attraversati (si noti che in questa notazione la condizione 3 equivale a prescrivere che non possano esserci punti ripetuti).

La *lunghezza* di un itinerario è data dal numero di tratti che lo compongono. Diciamo che (a,b) e (c,d) sono *collegati* se esiste almeno un itinerario da (a,b) a (c,d).

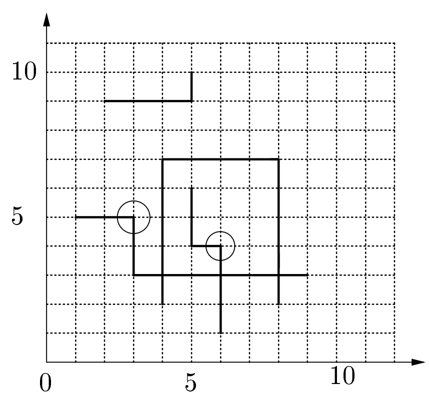
Dati due punti (a,b) e (c,d) del piano, la loro *separazione* consiste nel rimuovere da ogni strada ogni tratto che appartiene a un itinerario da (a,b) a (c,d). Si noti che l'operazione di separazione può modificare le strade del piano, in quanto una strada può essere accorciata o addirittura spezzata in due nuove strade. Le strade prodotte a seguito della separazione possono essere descritte ancora come $\sigma((x,y),\beta)$, dove β è una sottosequenza di una qualche sequenza α descrivente una strada esistente prima della separazione e (x,y) è il nuovo punto di origine.

Esempio

Supponiamo di aver inserito le strade:

$(1,5)\,EESSEEEEEE$ $\qquad$ $(8,2)\,NNNNNWWWWSSSSS$ $\qquad$ $(5,6)\,SSESSS$ $\qquad$ $(5,10)\,SWWW$

Le strade inserite sono visualizzate nella figura sottostante, dove vengono messi in evidenza i punti $(3,5)$ e $(6,4)$.



Esistono due itinerari fra $(3,5)$ e $(6,4)$: l'itinerario

$$(3,5),(3,4),(3,3),(4,3),(5,3),(6,3),(6,4)$$

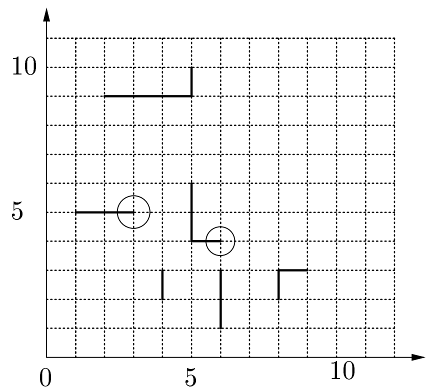
di lunghezza 6 e l'itinerario

$$\begin{array}{l}(3,5),(3,4),(3,3),(4,3),(4,4),(4,5),(4,6),(4,7),(5,7),(6,7),\\(7,7),(8,7),(8,6),(8,5),(8,4),(8,3),(7,3),(6,3),(6,4)\end{array}$$

di lunghezza 18. Per separare $(6,4)$ e $(3,5)$ occorre cancellare dalle strade del piano ogni tratto che appartiene ad almeno uno dei due itinerari. Dopo la separazione di $(6,4)$ e $(3,5)$, le strade presenti nel piano sono descritte da

$$(1,5)\,EE \qquad (8,3)\,E \qquad (8,2)\,N \qquad (4,3)\,S \qquad (5,6)\,SSE \qquad (6,3)\,SS \qquad (5,10)\,SWWW$$

rappresentate nella figura qui sotto.



Più in dettaglio, la prima strada costruita viene spezzata e dà origine alle strade $(1,5)\,EE$ e $(8,3)\,E$; la seconda strada produce le strade $(8,2)\,N$ e $(4,3)\,S$; la terza strada si spezza nelle strade $(5,6)\,SSE$ e $(6,3)\,SS$; la quarta strada rimane inalterata.

L'obiettivo del progetto consiste nel determinare alcune caratteristiche di strade e itinerari. In particolare, si richiede di implementare una struttura dati efficiente che permetta di eseguire le operazioni descritte sotto.

Si tenga presente che la dimensione della più piccola area rettangolare contenente tutte le strade contiene in genere un numero di punti interi di gran lunga superiore rispetto al numero di punti appartenenti alle strade. Pertanto, *non* è sicuramente efficiente rappresentare il piano mediante una matrice o strutture analoghe che memorizzino *ogni* punto del piano. Si devono inoltre evitare procedure che effettuano ricerche su *tutto* il piano.

- **crea**(x, y, α)

  Considera la strada $\sigma((x,y),\alpha)$. Se la strada contiene cicli (ossia, la strada percorre più di una volta lo stesso punto) viene stampato su una nuova linea il messaggio di errore

  `La strada` $(x,y)\,\alpha$ `contiene cicli`

  Altrimenti la strada viene costruita.

- **distanza**(a, b, c, d)

  Se (a,b) e (c,d) non sono collegati, stampa su nuova linea il messaggio:

```
I punti (a,b) e (c,d) non sono collegati
```

Altrimenti, calcola la minima fra le lunghezze di tutti gli itinerari da (a,b) a (c,d). Detto l tale valore, stampa su una nuova linea il messaggio:

```
La minima distanza fra (a,b) e (c,d) e' l
```

- **separa**(a, b, c, d)
  Effettua la separazione di (a, b) e (c, d).
- **mappa()**
  Stampa su una nuova linea il messaggio

  ```
  Strade presenti nel piano:
  ```

  seguito dall'elenco delle strade nel piano. Ogni strada viene visualizzata su una nuova linea nel formato (x, y) β (l'ordine in cui le strade vengono stampate è irrilevante).

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 α è una stringa sull'alfabeto $\{N, E, S, W\}$ e a, b, c, d sono interi. Quando una linea è letta viene eseguita l'operazione ad essa associata e viene stampato l'eventuale output prodotto dall'esecuzione dell'operazione associata; tutte le operazioni di stampa sono effettuate sullo standard output (`stdout`) e ogni operazione deve iniziare su una nuova linea.

LINEA DI INPUT	OPERAZIONE
`c` a b α	**crea** (a, b, α)
`d` a b c d	**distanza** (a, b, c, d)
`s` a b c d	**separa** (a, b, c, d)
`m`	**mappa** ()
`f`	Termina l'esecuzione del programma

Tabella 1: Specifiche del programma

Si noti che non devono essere presenti vincoli sulle dimensioni di strade e itinerari. 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()`.

Esempio

Si supponga che le linee di input siano:

```
c 2 3 NEEENNWWSSSWSEEE
c -3 -3 NNNNEESSSEEE
c 1 2 WWWSWSSEESS
c 0 3 SSSSSS
d 0 -3 -1 -3
s  0 -3 -1 -3
m
c 1 -3 NNWWWWWW
c 1 2 WWWSSSSSWW
d -3 -3 -1 -1
d -3 -3 1 2
s -3 -3 1 2
m
d -3 -3 -1 -1
f
```

L'output prodotto dal programma deve essere:

```
La strada  (2, 3) NEEENNWWSSSWSEEE contiene cicli
La minima distanza fra (0,-3) e (-1,-3) e' 3
Strade presenti nel piano:
(-3, -3) NN
(0, -2) EE
(1, 2) W
(0,3) S
La minima distanza fra (-3,-3) e (-1,-1) e' 4
La minima distanza fra (-3,-3) e (1,2) e'  9
Strade presenti nel piano:
(0, -2) EE
(0,3) S
(1 -3) NNWWW
(-3,-1) WW
(-3,-3) W
I punti (-3,-3) e (-1,-1) non sono collegati
```

Presentazione del progetto

Il progetto deve essere inviato per posta elettronica all'indirizzo `aguzzoli@dsi.unimi.it` entro il 28 Febbraio 2003. La discussione del progetto si svolgerà il 4 Marzo 2003 in aula 4 alle 9:00. L'esame orale si svolgerà l'11 Marzo 2003 in aula 4 alle 9:00.

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.

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 sempre entro il 28 Febbraio 2003 (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 S211.

Avvisi

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

Lo svolgimento del progetto è una prova d'esame da svolgere *individualmente*.