Laboratorio di Algoritmi e Strutture Dati

Docenti: M. Goldwurm, S. Aguzzoli

Appello del 7 febbraio 2006

Progetto "Mappe 2"
Consegna entro il 26 febbraio 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 intero, denotato con $Val(P)$.

Un *percorso* $\sigma(P_0, P_n)$ da P_0 a P_n è dato da una sequenza di punti $(P_0, \ldots, P_n)$ tali che, per ogni $1 \leq i \leq n-1$, posto $P_i = (x_i, y_i)$ e $P_{i+1} = (x_{i+1}, y_{i+1})$, vale $|x_i - x_{i+1}| + |y_i - y_{i+1}| = 1$.

La *lunghezza* $Length(\sigma)$ di un percorso σ è data dal numero di punti in σ. Il *valore* $Val(\sigma)$ di un percorso $\sigma = (P_0, \ldots, P_n)$ è dato dalla somma dei valori dei punti di σ, quindi:

$$Val(\sigma) \; = \; \sum_{i=0}^{n} Val(P_i)$$

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 il percorso $(P_0, \ldots, P_n)$ tale 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:

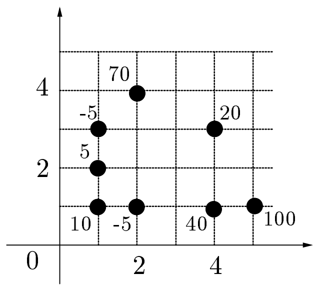
  $P_{k+1} = (x_k, y_k + 1)$ se $s_{k+1} = N$ (spostamento verso Nord);
  $P_{k+1} = (x_k + 1, y_k)$ se $s_{k+1} = E$ (spostamento verso Est);
  $P_{k+1} = (x_k, y_k - 1)$ se $s_{k+1} = S$ (spostamento verso Sud);
  $P_{k+1} = (x_k - 1, y_k)$ se $s_{k+1} = W$ (spostamento verso Ovest).

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. Il *valore* $Val(M)$ di una mappa M è dato dal valore di $\pi(M)$.

Un percorso $\tau(P,Q)$ da P a Q è *ottimale* se e solo se per ogni percorso $\sigma(P,Q)$ da P a Q vale una delle seguenti condizioni

1. $Length(\tau) < Length(\sigma)$
2. $Length(\tau) = Length(\sigma)$ e $Val(\tau) \geq Val(\sigma)$.

Esempio 1
Supponiamo che i punti del piano abbiano i valori indicati nella figura (i punti senza alcuna indicazione hanno valore 0).



Supponiamo di cercare un percorso ottimale τ_1 da $P = (1,1)$ a $Q = (4,3)$. Tale percorso è un percorso di valore massimo fra i percorsi da P a Q di lunghezza 6 (minima lunghezza di un percorso da P a Q). Esiste un unico percorso ottimale, ossia

$$\tau_1 \;=\; (1,1),\,(2,1),\,(3,1)\,(4,1),\,(4,2),\,(4,3)$$

di valore 65. Sia R il punto $(3,3)$. Da P a R ci sono due percorsi ottimali

$$\begin{aligned} \tau_2 &= (1,1),\,(1,2),\,(2,2)\,(3,2),\,(3,3) \\ \tau_3 &= (1,1),\,(1,2),\,(2,2)\,(2,3),\,(3,3) \end{aligned}$$

di valore 15. Da $(4,2)$ a P c'è un unico percorso ottimale

$$(4,2),\,(4,1),\,(3,1),\,(2,1)\,(1,1)$$

di valore 45.

Dato un percorso $\sigma(P,Q)$ da P a Q la *riduzione di* σ è un percorso ottimale $\tau(P,Q)$. 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.

Dato un percorso $\sigma = (P_0, \ldots, P_n)$, e due indici $0 \leq i \leq j \leq n$, il *sottopercorso* $\sigma_{i,j}$ di σ è la sequenza di punti $(P_i, P_{i+1}, \ldots, P_j)$. Sia Σ l'insieme di tutti i sottopercorsi di σ. Il *miglior sottopercorso* di σ è un sottopercorso $\sigma_{h,k} \in \Sigma$ di valore V massimo, indice h minimo e lunghezza massima fra quelli di valore V e indice minimo h. Formalmente, detto V il valore $Val(\sigma_{h,k})$:

- V è il massimo tra i valori dei sottopercorsi di σ:

$$V \;=\; \max\{Val(\sigma_{i,j}) \mid \sigma_{i,j} \in \Sigma\}\,.$$

- Sia $\Gamma = \{\sigma_{i,j} \in \Sigma \mid Val(\sigma_{i,j}) = V\}$ e sia $h = \min\{i \mid \sigma_{i,j} \in \Gamma\}$. Allora, $k \;=\; \max\{j \mid \sigma_{h,j} \in \Gamma\}$.

Esempio 2
Si supponga che il percorso σ contenga 14 punti $(P_0, \ldots, P_{13})$ i cui punti hanno i valori

P_0	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}
-5	1	3	-4	10	0	-20	10	-7	-1	2	3	3	-1

Il massimo valore V di un sottopercorso di σ è 10. I sottopercorsi di valore 10 sono:

$$\sigma_{1,4} \qquad \sigma_{1,5} \qquad \sigma_{4,4} \qquad \sigma_{4,5} \qquad \sigma_{7,7} \qquad \sigma_{7,12}$$

Il miglior sottopercorso è quindi $\sigma_{1,5}$.

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** $(m, n, nome)$

  Elimina tutte le mappe attualmente esistenti e crea un nuovo piano, assegnando a ogni punto del piano i valori corrispondenti ai parametri $m > 0$, $n > 0$ e *nome*, come descritto nella nota 3.

- **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.

- **riduci** (w)

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

- **sottopercorso** (w)

  Stampa il miglior sottopercorso di $\pi(M)$, dove M è la mappa di nome w, secondo il formato specificato nella nota 4. Se non esiste alcuna mappa di nome w non esegue alcuna operazione.

- **elimina** (w)

  Elimina la mappa di nome w. Se non esiste alcuna mappa di nome w non esegue alcuna operazione.

- **stampa** (w)

  Stampa su una nuova riga l'origine e la specifica della mappa di nome w secondo il formato specificato nota 5. Se non esiste alcuna mappa di nome w non esegue alcuna operazione.

- **lista** ()

  Stampa tutti le mappe esistenti secondo il formato specificato nella nota 6.

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 x, y, z sono interi, m e n sono interi positivi, w è una stringa non vuota sull'alfabeto $\mathcal{A}$, α è una stringa non vuota sull'alfabeto $\{N, E, S, W\}$ e *nome* è una stringa non vuota corrispondente al nome di un file.

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
`c` $m\ n\ nome$	**crea** $(m, n, nome)$
`v` $x\ y\ z$	**valore** (x, y, z)
`m` $w\ x\ y\ \alpha$	**mappa** (w, x, y, α)
`r` w	**riduci** (w)
`S` w	**sottopercorso** (w)
`s` w	**stampa** (w)
`e` w	**elimina** (w)
`l`	**lista** ()
`f`	Termina l'esecuzione del programma

Tabella 1: Specifiche del programma

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()`.

3. Dati un intero x e un intero $q > 0$, denoteremo con $x \bmod q$ il resto non negativo r della divisione (intera) $x : q$. Si noti che $0 \leq r < q$. Ad esempio: $12 \bmod 7 = 5$ e $-2 \bmod 7 = 5$.

   Si consideri il comando `c` m n *nome*. Il file di nome "nome" conterrà la sequenza di $m \cdot n$ interi $v_0, v_1, \ldots, v_{m \cdot n-1}$. Tali interi determineranno i valori di ogni punto del piano secondo la prescrizione seguente: Se $a = x \bmod m$ e $b = y \bmod n$ allora $Val(x, y) = v_{a+mb}$.

4. $M = \langle w, P, \alpha\rangle$ una mappa presente nel piano, e sia $\sigma = ((x_1, y_1), (x_2, y_2), \ldots, (x_u, y_u))$ il miglior sottopercorso di $\pi(M)$. Sia V il valore di σ. L'output del comando `S` w va visualizzato nel seguente formato:

   ```
   V: x1,y1; x2,y2; ... xu,yu
   ```

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

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

6. 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 righe di input siano:

```
c 4 4 primo.txt
m mappa1 4 7 EEESSSEENWWWWWWWSSEENNNWSSSSSWW
l
S mappa1
r mappa1
s mappa1
v 1 7 10
r mappa1
s mappa1
l
m mappa1 4 7 EEESSSEENWWWWWWWSSEENNNWSSSSSWW
l
v 3 2 18
v 4 2 1
S mappa1
r mappa1
s mappa1
c 1 1 secondo.txt
e mappa1
l
m mappa1 0 0 EEENNWWWSSEEENN
S mappa1
v 1 0 -2
v 1 2 -2
```

```
v 2 2 0
r mappa1
s mappa1
S mappa1
c 4 3 terzo.txt
m alfa -6 3 WSEEEENNNNESW
m beta 4 3 EENNWWSSEEEESS
S alfa
v -7 2 -7
S alfa
v -6 2 7
S alfa
r alfa
s alfa
v -3 3 100
r alfa
s alfa
S beta
v 6 5 -5
S beta
r beta
s beta
v 4 2 -2
r beta
s beta
v 6 3 0
S beta
l
f
```

dove il contenuto del file `primo.txt` è il seguente:

```
-1 -3 2 -2 2 1 0 2 -2 3 2 -1 0 -2 1 -2
```

il contenuto del file `secondo.txt` è il seguente:

```
-1
```

e il contenuto del file `terzo.txt` è il seguente:

```
-4 6 1 -5 3 10 -2 7 2 -8 4 -1
```

L'output prodotto dal programma deve essere:

```
(
mappa1 4 7 1 EEESSSEENWWWWWWWSSEENNNWSSSSSWW
)
13: 9,5; 8,5; 7,5; 6,5; 5,5; 4,5; 3,5; 2,5; 2,4; 2,3
mappa1 4 7 11 SSWWSSSWS
mappa1 4 7 12 WWWSSSSSS
(
```

```
mappa1 4 7 12 WWWSSSSSS
)
(
mappa1 4 7 1 EEESSSEENWWWWWWWSSEENNNWSSSSSWW
)
28: 9,5; 8,5; 7,5; 6,5; 5,5; 4,5; 3,5; 2,5; 2,4; 2,3; 3,3; 4,3; 4,4; 4,5; 4,6; 3,6;
3,5; 3,4; 3,3; 3,2; 3,1; 2,1; 1,1
mappa1 4 7 24 SSSSSWWWS
(
)
-1: 0,0
mappa1 0 0 -5 NEENE
0: 2,2
19: -3,3; -3,4; -3,5; -3,6; -2,6; -2,5
19: -3,3; -3,4; -3,5; -3,6; -2,6; -2,5
19: -6,3; -7,3; -7,2; -6,2; -5,2; -4,2; -3,2; -3,3; -3,4; -3,5; -3,6; -2,6; -2,5
alfa -6 3 11 NEEEN
alfa -6 3 94 EEENN
9: 5,3; 6,3; 6,4; 6,5; 5,5; 4,5; 4,4; 4,3; 5,3; 6,3
8: 4,5; 4,4; 4,3; 5,3; 6,3
beta 4 3 19 SSEEEE
beta 4 3 16 EESESE
19: 5,3; 6,3; 6,2; 7,2; 7,1; 8,1
(
alfa -6 3 94 EEENN
beta 4 3 15 EESESE
)
```

Presentazione del progetto

Il progetto deve essere inviato per posta elettronica all'indirizzo `aguzzoli@dsi.unimi.it` entro il 26 Febbraio 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 27 Febbraio 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.