Laboratorio di Algoritmi e Strutture Dati

Docenti: M. Torelli, S. Aguzzoli

Appello dell'8 Febbraio 2007

Progetto "Asteroidi"
Consegna entro il 27 Febbraio 2007

Il problema

Due pianeti ruotano attorno al loro sole in un unico piano orbitale e sono collegati tramite un servizio di astronavi di linea. Tra i due pianeti è interposta una fascia di asteroidi, che le astronavi devono attraversare in ogni viaggio.

Per pianificare il viaggio in modo tale da ridurre l'usura determinata dall'attraversamento della fascia di asteroidi, le astronavi si avvalgono dei dati forniti da una rete di satelliti collocati nella fascia stessa.

Formalmente, il piano orbitale è l'insieme dei punti $(x, y) \in \mathbb{Z}^2$. Dato un punto $(x_0, y_0) \in \mathbb{Z}^2$ e due interi positivi r e c, una *regione rettangolare* $\mathit{Rett}((x_0, y_0), r, c)$ è il rettangolo il cui vertice in alto a sinistra è (x_0, y_0) e la cui base e altezza contengono rispettivamente c e r punti (i punti nel rettangolo possono essere visti come gli elementi di una matrice $r \times c$). Più precisamente:

$$\mathit{Rett}((x_0, y_0), r, c) \;=\; \{\, (x, y) \in \mathbb{Z}^2 \;\;|\;\; x_0 \leq x < x_0 + c,\; y_0 - r < y \leq y_0 \,\}\,.$$

Un *satellite* sorveglia una regione rettangolare della fascia contenente gli asteroidi e associa a ogni punto del rettangolo un numero intero. A ogni satellite è associata come *codice identificativo* una stringa finita di lunghezza arbitraria sull'alfabeto $\{A, B, \ldots, Z\}$ delle lettere maiuscole. Formalmente, un satellite è definito dalla quintupla

$$S \;=\; \langle\, (x_S, y_S),\, r,\, c, M_S,\, \sigma_S \,\rangle$$

dove:

- $(x_S, y_S) \in \mathbb{Z}^2$, r e c sono interi positivi; tali valori definiscono la regione rettangolare $\mathit{Rett}((x_S, y_S), r, c)$ sorvegliata dal satellite.
- M_S è una matrice $r \times c$ di interi che definisce i valori dei punti in $\mathit{Rett}((x_S, y_S), r, c)$ nel modo sotto specificato.
- σ_S è il codice identificativo di S.

Satelliti distinti hanno codici distinti, vale a dire che se $S_1 \neq S_2$ allora $\sigma_{S_1} \neq \sigma_{S_2}$. La matrice M_S determina una mappatura del piano $\mathcal{P}_S : \mathbb{Z}^2 \to \mathbb{Z}$ definita da:

$$\mathcal{P}_S(x, y) \;=\; \begin{cases} M_S[y_S - y, x - x_S] & \text{se } 0 \leq y_S - y < r \text{ e } 0 \leq x - x_S < c \\ 0 & \text{altrimenti} \end{cases}$$

dove $M_S[i, j]$ denota il valore della matrice M_S alla riga i ($0 \leq i < r$) e colonna j ($0 \leq j < c$). Ad esempio, se $(x_S, y_S) = (-3, 5)$ e M_S è una matrice 10×20, allora $\mathcal{P}_S(-3, 5) = M_S[0, 0]$, $\mathcal{P}_S(-2, 5) = M_S[0, 1]$, $\mathcal{P}_S(-3, 4) = M_S[1, 0]$, mentre $\mathcal{P}_S(0, -5) = 0$ e $\mathcal{P}_S(17, 4) = 0$.

Per decidere la rotta da seguire nell'attraversamento della fascia di asteroidi, un'astronave prende in considerazione una rete di satelliti e costruisce a partire da questi una mappa del piano orbitale.

Una *rete* di satelliti R è un insieme finito $R = \{S_1, S_2, \ldots, S_n\}$ di satelliti. La *mappa* $\mathcal{P}_R$ associata alla rete R è la funzione $\mathcal{P}_R : \mathbb{Z}^2 \to \mathbb{Z}$ tale che

$$\mathcal{P}_R(x, y) \ = \ \sum_{S \in R} \mathcal{P}_S(x, y)\,.$$

$\mathcal{P}_R(x, y)$ è il *coefficiente di transitabilità* del punto (x, y) stimato dalla rete R. Più alto è il valore di $\mathcal{P}_R(x, y)$, minore è l'usura subita da un'astronave che transita per (x, y).

Data una regione rettangolare $T = \mathit{Rett}((x_0, y_0), r, c)$, e due interi x_c, k tali che $x_0 \leq x_c < x_0 + c$ e $0 < k \leq c - (x_c - x_0)$, il *corridoio di attraversamento* $C_T(x_c, k)$ di T è il rettangolo $\mathit{Rett}((x_c, y_0), r, k)$. Si noti che, per i vincoli posti su x_c e k, i punti di $C_T(x_c, k)$ sono contenuti in T e i due rettangoli hanno la stessa altezza. Chiamiamo k l'*ampiezza* del corridoio $C_T(x_c, k)$. Dato un corridoio C di un rettangolo $\mathit{Rett}((x_0, y_0), r, c)$ e una rete di satelliti R, il coefficiente di transitabilità $\mathcal{T}_R(C)$ del corridoio C rispetto alla rete R è dato dalla somma dei valori $\mathcal{P}_R(x, y)$ dei punti (x, y) di C:

$$\mathcal{T}_R(C) \ = \ \sum_{(x,y) \in C} \mathcal{P}_R(x, y)\,.$$

Data una regione rettangolare del piano orbitale da superare e una rete R di satelliti, un'astronave seleziona in essa un corridoio di attraversamento C in modo tale da massimizzare il valore d $\mathcal{T}_R(C)$ e l'ampiezza del corridoio. Più precisamente, dati:

- una rete di satelliti R;
- una regione $T = \mathit{Rett}((x, y), r, c)$;

sia M il massimo valore del coefficiente di transitabilità di un corridoio C di T, ossia:

$$M \ = \ \max\{\ \mathcal{T}_R(C) \ \mid \ C \text{ è un corridoio di } \mathit{Rett}((x, y), r, c)\ \}$$

Allora il corridoio selezionato sarà un corridoio C del rettangolo T di ampiezza massima fra i corridoi di T tali che $\mathcal{T}_R(C) = M$. Questo significa che, se C ha ampiezza k e se C' è un altro corridoio di T tale che $\mathcal{T}_R(C') = M$, allora, detta k' l'ampiezza di C', deve valere $k' \leq k$.

Esempio

Si supponga di avere definito i satelliti

$$S_1 \ = \ \langle(2,5),\, 3,\, 2, M_1,\, A\rangle \qquad S_2 \ = \ \langle(4,5),\, 2,\, 4, M_2,\, B\rangle \qquad S_3 \ = \ \langle(6,5),\, 3,\, 1, M_3,\, C\rangle$$

dove le matrici sono:

$$M_1 \ = \ \begin{bmatrix} 1 & -2 \\ 3 & 6 \\ -1 & 4 \end{bmatrix} \qquad M_2 \ = \ \begin{bmatrix} -17 & -1 & 10 & -2 \\ 5 & 5 & 1 & 6 \end{bmatrix} \qquad M_3 \ = \ \begin{bmatrix} -4 \\ 1 \\ -1 \end{bmatrix}$$

Consideriamo la rete di satelliti $R = \{S_1, S_2, S_3\}$. La mappa $\mathcal{P}_R$ associata alla rete R è descritta nella Figura 1, dove vengono evidenziati solamente i punti (x, y) tali che $\mathcal{P}_R(x, y) \neq 0$. Consideriamo il rettangolo $T_1 = \mathit{Rett}((1, 5), 3, 5)$. Il massimo valore $\mathcal{T}_R(C)$ che può assumere un corridoio C di T_1 è 11. Precisamente, esistono due corridoi C_1 e C_2 con coefficiente di transitabilità 11:

$$C_1 = C(1, 3) \qquad C_2 = C(2, 2)$$

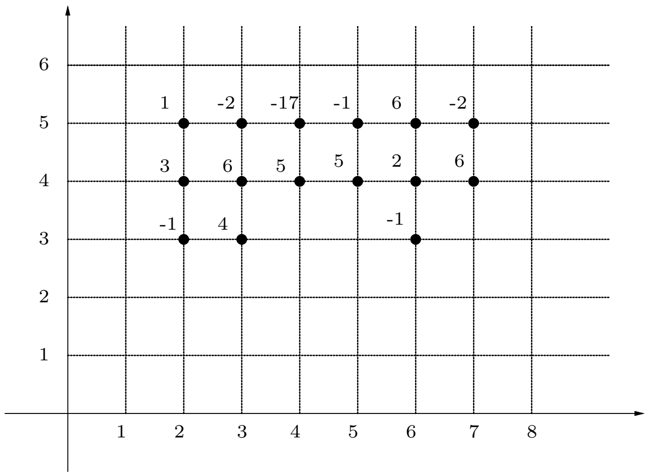

Figura 1: La mappa $\mathcal{P}_R$

dove C_1 corrisponde al rettangolo $Rett((1,5),3,3)$ e C_2 corrisponde al rettangolo $Rett((2,5),3,2)$. Essendo l'ampiezza di C_1 maggiore di quella di C_2, il corridoio selezionato è C_1.
Consideriamo il rettangolo $T_2 = Rett((1,5),3,6)$. Anche in questo caso il massimo valore del coefficiente di transitabilità di un corridoio di T_2 è 11. I corridoi con coefficiente di transitabilità massimo sono

$$C_1 = C(1,3) \qquad C_2 = C(2,2) \qquad C_3 = C(5,2)$$

Il corridoio selezionato è C_1.
Sia $T_3 = Rett((1,5),3,7)$. In questo caso il massimo valore di $\mathcal{T}_R(C)$ è 15, corrispondente al corridoio $C_4 = C(5,3)$, quindi il corridoio selezionato è C_4.
Infine, sia $T_4 = Rett((3,5),2,3)$. I corridoi di T_4 aventi coefficiente di transitabilità massimo sono

$$C_5 = C(3,1) \qquad C_6 = C(5,1)$$

Poiché C_5 e C_6 hanno la stessa ampiezza, può essere selezionato indifferentemente uno dei due corridoi.

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 satelliti può essere molto grande rispetto al numero di satelliti presenti nel piano orbitale, quindi *non è sicuramente efficiente rappresentare il piano mediante un*'unica *matrice*).

- **satellite** $(x, y, r, c, file, \sigma)$

  Se non esiste già un satellite di codice σ, crea il satellite $S = \langle (x_S, y_S), r, c, M_S, \sigma_S \rangle$, dove $x_S = x$, $y_S = y$, $\sigma_S = \sigma$ e i valori di M_S sono letti dal file di nome $file$ secondo il formato riportato nell'apposita sezione (se il file non esiste, viene solamente stampato un opportuno messaggio di errore). Se esiste già un satellite di codice σ non compie alcuna operazione.

- **aggiorna** $(\sigma, file)$

  Se non esiste già un satellite S di codice σ, non compie alcuna operazione. Altrimenti sostituisce i dati forniti dalla matrice M_S con quelli letti dal file di nome $file$ secondo il formato riportato nell'apposita sezione (se il file non esiste, viene solamente stampato un opportuno messaggio di errore).

- **rete** $(\sigma_1, \sigma_2, \ldots, \sigma_n)$

  Se per qualche $i = 1, \ldots, n$ non esiste alcun satellite di codice σ_i allora non compie alcuna operazione. Altrimenti distrugge l'eventuale rete esistente e crea una nuova rete $R = \{S_1, S_2, \ldots, S_n\}$, dove $\sigma_{S_i} = \sigma_i$ per ogni $i = 1, \ldots, n$.

- **transitabilità** (x, y)

  Se non esiste alcuna rete non compie alcuna operazione. Altrimenti stampa il valore del coefficiente di transitabilità nel punto (x, y) stimato dalla rete esistente.

- **corridoio** (x_0, y_0, r, c)

  Se non esiste alcuna rete non compie alcuna operazione. Altrimenti stampa, secondo il formato riportato nell'apposita sezione, un corridoio di attraversamento della regione $Rett((x_0, y_0), r, c)$.

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 $file$ è il nome di un file, $\sigma, \sigma_1, \sigma_2, \ldots, \sigma_n$ sono stringhe finite sull'alfabeto $\{A, B, \ldots, Z\}$ di lunghezza arbitraria, x e y sono interi, r,c sono interi positivi. 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
`s` $x\ y\ r\ c\ file\ \sigma$	**satellite** $(x, y, r, c, file, \sigma)$
`a` $\sigma\ file$	**aggiorna** $(\sigma, file)$
`r` $\sigma_1\ \sigma_2\ \cdots\ \sigma_n$	**rete** $(\sigma_1, \sigma_2, \ldots, \sigma_n)$
`t` $x\ y$	**transitabilità** (x, y)
`c` $x\ y\ r\ c$	**corridoio** (x, y, r, c)
`f`	Termina l'esecuzione del programma

Tabella 1: Specifiche del programma

Note

1. Non devono essere presenti vincoli sul numero e sulla dimensione dei rilevamenti (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. **Formato per la rappresentazione di una matrice**

   (a) Si supponga che la riga in input sia

   $$\texttt{s}\ \ x\ \ y\ \ r\ \ c\ \ file\ \ \sigma$$

   Allora il file di nome $file$ deve contenere $r \cdot c$ numeri interi $a_0, a_1, \ldots, a_{r \cdot c-1}$, separati fra loro da uno o più caratteri di spaziatura (spazio, a capo, etc.), che rappresentano i valori della matrice M_S di dimensione $r \times c$, associata al satellite S: per ogni $0 \leq i < r$ e $0 \leq j < c$, il valore di $M_S[i, j]$ è dato dal numero $a_{c \cdot i+j}$ contenuto in $file$ alla posizione $c \cdot i + j$ (nota: il primo numero del file è in posizione 0).

   (b) Si supponga ora che la riga in input sia

   $$\texttt{a}\ \ \sigma\ \ filenuovo$$

   e che il satellite di nome σ sia stato precedentemente introdotto con un comando

   $$\texttt{s}\ \ x\ \ y\ \ r\ \ c\ \ filevecchio\ \ \sigma$$

   Allora il file di nome $filenuovo$ deve contenere $r \cdot c$ numeri interi $a_0, a_1, \ldots, a_{r \cdot c-1}$, separati fra loro da uno o più caratteri di spaziatura (spazio, a capo, etc.), che rappresentano i valori *aggiornati* della matrice M_S di dimensione $r \times c$, associata al satellite S. I valori del file sono associati alla matrice come spiegato al punto (a).

4. **Formato per la stampa del corridoio scelto**

   Si supponga che il corridoio selezionato in base al comando

   $$\texttt{c}\ \ x\ \ y\ \ r\ \ c$$

   sia $C(x_c, k)$. Allora si deve stampare, su una nuova riga,

   $$x\ \ y\ \ r\ \ c\ \ :\ \ x_c\ \ k$$

Esempio

Si supponga che le righe di input siano:

```
s -8 -3  4  3  alfa.txt  ALFA
s -4  0  2  8  beta.txt  BETA
s  7  6 16  3 gamma.txt GAMMA
s  1 -1  2  8 delta.txt DELTA
r ALFA BETA
t -6 -3
c -8 -1 5 13
c -8 0 5 13
c -1000 1000 1006 1100
a BETA beta1.txt
c -1000 1000 1006 1100
r BETA GAMMA DELTA
t -6 -3
c -1000 1000 1006 1100
f
```

dove il file `alfa.txt` contiene i valori

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

il file `beta.txt` contiene i valori

```
 1 -2  5  1  1  1  1  2
 3 -4  4 -3 -3 -3  4  4
```

il file `beta1.txt` contiene i valori

```
 1 -2  5  1  1  1  1   2
 3 -4  4  0 -3 -3  4 -10
```

il file `gamma.txt` contiene i valori

```
 -1   2   0
  4  -1  -1
  3   9   4
 -1   7   3
  5 -10   4
 -1  -5  -7
  6   4  -8
  4   2   5
  4   2  -8
  4  -3  -8
-11   1  -8
-10   4   1
  6   6   6
  6   6   6
  6   6   6
  7   7   7
```

e il file `delta.txt` contiene i valori

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

L'output prodotto dal programma deve essere:

```
4
-8 -1 5 13 : -7 6
-8 0 5 13 : -8 13
-1000 1000 1006 1100 : -7 107
-1000 1000 1006 1100 : -7 10
0
-1000 1000 1006 1100 : -2 2
```

Presentazione del progetto

Il progetto deve essere inviato per posta elettronica all'indirizzo `aguzzoli@dsi.unimi.it` entro il 27 Febbraio 2007 (incluso).

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 2007 (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.

La discussione del progetto e l'esame orale di Algoritmi e Strutture Dati si svolgeranno **indicativamente** nei giorni 2, 6, 9, 12 Marzo 2007.

Alla consegna del progetto, indicare nel *testo* della e-mail la data in cui si preferisce sostenere la prova orale; nei limiti del possibile si cercherà di tener conto di tali indicazioni (se non si hanno preferenze, non dare alcuna indicazione).

Il calendario degli esami orali sarà disponibile sulla pagina del corso qualche giorno dopo il termine di consegna del progetto.

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.