

Laboratorio di Algoritmi e Strutture Dati

Docente: S. Aguzzoli

Progetto “Stonehenge 2”

valido per l'appello di settembre 2013

1 Il problema

Una civiltà neolitica erige complessi megalitici allo scopo di immaginare cosa ne fantasticheranno i posteri. A questo fine la popolazione di tale civiltà utilizza delle cave di roccia presenti nel territorio, vie di comunicazione come strade e canali per il trasporto di blocchi di roccia, siti cerimoniali dove erigere e posizionare i blocchi stessi.

1.1 Trasportare i blocchi

Il trasporto dei blocchi dalle cave avviene caricandoli su zattere che navigano in canali artificiali. Tali canali sono composti da tratte rettilinee e servono un insieme di cave. Per semplicità supponiamo che una *tratta* di canale sia allineata in direzione est-ovest. Ogni *cava* dell'insieme servito dalla tratta è collegata al canale attraverso una *strada di servizio* perpendicolare alla tratta. Siamo interessati alle distanze da coprire lungo queste strade.

Possiamo rappresentare il territorio con un sistema di assi cartesiane dove l'asse delle ascisse è allineato da ovest a est, mentre quello delle ordinate da sud a nord.

A ogni insieme di cave $\mathcal{C} = \{c_1, c_2, \dots, c_n\}$ servito da una determinata tratta è associata una funzione *posizione* $p: \mathcal{C} \rightarrow \mathbb{Z} \times \mathbb{Z}$, dove $p(c_i) = (x_i, y_i)$ rappresenta la posizione della cava c_i nel territorio.

Per semplicità, assumiamo sempre che se $i \neq j$ allora $x_i \neq x_j$.

La tratta del canale giace su una retta parallela all'asse delle ascisse, di ordinata $y \in \mathbb{Z}$. La *percorrenza totale* \mathcal{P} della tratta è la somma delle lunghezze delle strade di servizio, ovvero:

$$\mathcal{P} = \sum_{i=1}^n |y_i - y|.$$

Dato un insieme di cave $\mathcal{C} = \{c_1, c_2, \dots, c_n\}$ la popolazione determina la posizione y della tratta in modo tale da minimizzare \mathcal{P} .

Esempio 1. Sia $\mathcal{C} = \{c_1, \dots, c_8\}$ l'insieme delle cave nel territorio, dove

$$\begin{aligned} p(c_1) &= (4, 7), & p(c_2) &= (18, 6), & p(c_3) &= (10, 4), & p(c_4) &= (-4, 3), \\ p(c_5) &= (-2, 6), & p(c_6) &= (-7, 8), & p(c_7) &= (6, 1), & p(c_8) &= (9, -3). \end{aligned}$$

Allora, una possibile scelta di ordinata y tale che la percorrenza totale della tratta è minima è $y = 4$, con $\mathcal{P} = 22$. Un'altra possibile scelta minima è $y = 5$, con, ovviamente, $\mathcal{P} = 22$ anche in questo caso. Una scelta di y che non corrisponde alla percorrenza minima è $y = 3$, con $\mathcal{P} = 24$.

1.2 Individuare i siti

Per ogni insieme $\mathcal{C} = \{c_1, c_2, \dots, c_n\}$ di cave servito da una determinata tratta la civiltà erige due *siti cerimoniali*, ovvero complessi megalitici costruiti con i blocchi trasportati lungo la tratta di canale. I

due siti sono posti in corrispondenza di due cave in \mathcal{C} (eventualmente coincidenti). In particolare le posizioni (x_a, y_a) e (x_b, y_b) dei due siti devono essere tali da minimizzare la somma totale delle lunghezze delle porzioni di canale necessarie a portare i blocchi di ogni cava di \mathcal{C} in almeno uno dei due siti. Equivalentemente, la somma

$$\mathcal{L} = \sum_{i=1}^n \min\{|x_i - x_a|, |x_i - x_b|\}$$

deve essere minima rispetto a ogni scelta di $(x_a, y_a), (x_b, y_b) \in \{p(c_i) \mid c_i \in \mathcal{C}\}$.

Esempio 2. Sia \mathcal{C} come nell'Esempio 1. Allora, le posizioni idonee per i due siti cerimoniali sono: $(x_a, y_a) = (-4, 3)$ e $(x_b, y_b) = (9, -3)$ per le quali risulta $\mathcal{L} = 23$. Una scelta che non minimizza \mathcal{L} è ad esempio $(x_a, y_a) = (4, 7)$ e $(x_b, y_b) = (18, 6)$, per la quale $\mathcal{L} = 38$.

1.3 Posizionare i blocchi

Ogni blocco consegnato a un sito cerimoniale deve essere posizionato nel complesso megalitico. Il blocco viene portato nella sua posizione finale attraverso una serie di mosse, chiamate *rotazioni*.

Un blocco $B(x, y, z)$ è un parallelepipedo i cui lati, paralleli alle direzioni degli assi del *sistema di riferimento del sito*¹, sono lunghi rispettivamente $x, y, z \in \mathbb{Z}$. La collocazione attuale di un blocco è determinata dalle coordinate del suo vertice di sud-ovest appoggiato sul terreno. Ovvero, se $(x_0, y_0, 0)$ sono le coordinate di tale vertice, allora la *collocazione* di $B(x, y, z)$ è:

$$S(x, y, z, x_0, y_0) = \{(r, s, t) \in \mathbb{Z}^3 \mid x_0 \leq r \leq x_0 + x, y_0 \leq s \leq y_0 + y, 0 \leq t \leq z\}.$$

Il blocco può essere ruotato in ognuna delle quattro direzioni cardinali e dunque portato in una nuova collocazione. Tali rotazioni sono descritte come segue:

- La rotazione *verso nord* di $B(x, y, z)$ in posizione $(x_0, y_0, 0)$ avviene attorno all'asse su cui giace il segmento di estremi $(x_0, y_0 + y, 0)$ e $(x_0 + x, y_0 + y, 0)$.
- La rotazione *verso sud* di $B(x, y, z)$ in posizione $(x_0, y_0, 0)$ avviene attorno all'asse su cui giace il segmento di estremi $(x_0, y_0, 0)$ e $(x_0 + x, y_0, 0)$.
- La rotazione *verso ovest* di $B(x, y, z)$ in posizione $(x_0, y_0, 0)$ avviene attorno all'asse su cui giace il segmento di estremi $(x_0, y_0, 0)$ e $(x_0, y_0 + y, 0)$.
- La rotazione *verso est* di $B(x, y, z)$ in posizione $(x_0, y_0, 0)$ avviene attorno all'asse su cui giace il segmento di estremi $(x_0 + x, y_0, 0)$ e $(x_0 + x, y_0 + y, 0)$.

Esempio 3. Si consideri il blocco $B(3, 4, 5)$ in posizione $(6, 7, 0)$. Allora la collocazione del blocco è $S(3, 4, 5, 6, 7) = \{(r, s, t) \mid 6 \leq r \leq 9, 7 \leq s \leq 11, 0 \leq t \leq 5\}$. La rotazione *verso nord* porta il blocco originale nel blocco $B(3, 5, 4)$ in posizione $(6, 11, 0)$. La rotazione *verso sud* porta il blocco in $B(3, 5, 4)$ in posizione $(6, 2, 0)$. La rotazione *verso ovest* porta il blocco in $B(5, 4, 3)$ in posizione $(1, 7, 0)$. La rotazione *verso est* porta il blocco in $B(5, 4, 3)$ in posizione $(9, 7, 0)$.

Un blocco $B(x, y, z)$ in posizione $(x_0, y_0, 0)$ può essere *liberamente ruotato* in una data direzione se la nuova collocazione del blocco non viene a sovrapporsi alle collocazioni dei blocchi già presenti. In dettaglio, se le collocazioni degli altri blocchi già presenti sono S_1, S_2, \dots, S_k , e se la nuova collocazione del blocco a seguito della rotazione è S , allora deve valere $S \cap S_i = \emptyset$ per ogni $i \in \{1, 2, \dots, k\}$.

Esempio 4. Supponiamo che nel sito siano già presenti i seguenti blocchi: $B(2, 6, 2)$ in posizione $(11, 1, 0)$, $B(4, 4, 7)$ in posizione $(4, -2, 0)$, $B(11, 3, 4)$ in posizione $(-2, 9, 0)$, e infine $B(1, 1, 5)$ in posizione $(6, 5, 0)$. Allora quest'ultimo blocco può essere liberamente ruotato solo verso ovest. In tutte le altre tre direzioni andrebbe a sovrapporsi alla collocazione di qualcuno degli altri blocchi.

¹Ovviamente, questo sistema di riferimento non ha nulla a che fare col sistema di riferimento che rappresenta le posizioni delle cave e dei siti, le tratte di canale e le strade nel territorio.

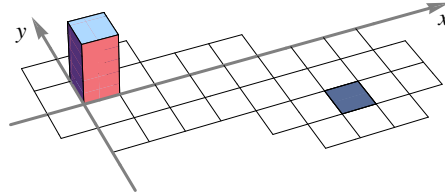


Figura 1: Un percorso cerimoniale per una stele.

1.4 La stele

Per completare un sito cerimoniale viene collocato nella sua sede finale un ultimo blocco, chiamato *stela*. La stele è un blocco $B(1, 1, 2)$. Tale blocco deve essere spostato da una *posizione iniziale* $(x_0, y_0, 0)$ a una *posizione finale* $(x_1, y_1, 0)$ attraverso una sequenza finita di mosse tali che:

1. Ogni mossa consiste in una rotazione del blocco.
2. La base² del blocco nella posizione iniziale e dopo ogni mossa deve essere totalmente contenuta in una specificata regione $K \subset \mathbb{Z} \times \mathbb{Z}$ di piano detta *percorso cerimoniale*.
3. La stele non può mai toccare, in alcuna mossa, eventuali blocchi collocati nel percorso cerimoniale. Tali blocchi vengono chiamati *ostacoli*.
4. Quando si trova in posizione finale, la stele deve avere collocazione $S(1, 1, 2, x_1, y_1)$.

Tale sequenza finita di mosse si chiama *cammino* della stele. Il numero di mosse che costituiscono un cammino è detto *lunghezza* del cammino.

Esempio 5. *Supponiamo che il percorso cerimoniale sia come indicato in Figura 1, la stele sia nella posizione iniziale $(0, 0, 0)$, mentre la posizione finale sia, come risulta dalla figura, in posizione $(6, -3, 0)$. Allora un cammino di lunghezza 8 è costituito dalla sequenza:*

$$S, E, E, E, E, E, E, S,$$

dove S indica una rotazione verso sud, mentre E indica una rotazione verso est. Un cammino di lunghezza 7 è invece fornito dalla sequenza:

$$E, E, S, E, E, E, S.$$

Non esistono cammini di lunghezza inferiore a 7. Supponiamo ora che il sito cerimoniale contenga un blocco $B(2, 7, 1)$ in posizione $(3, -3, 0)$. Allora non esiste alcun cammino per la stele.

2 Specifiche di progettazione

Si deve progettare un programma che sia in grado di gestire i dati relativi a blocchi, cave, tratte, siti, e quant'altro e di eseguire le seguenti operazioni. Il progetto del programma va illustrato in una relazione scritta.

²La base di un blocco in una certa collocazione, è l'insieme dei punti nell'intersezione fra la collocazione stessa e il piano di equazione $z = 0$.

La progettazione deve prevedere la scelta delle strutture dati da usare per rappresentare i dati e gli algoritmi da applicare per risolvere in maniera efficiente i problemi descritti nella traccia (non si devono assumere limiti su numeri di blocchi, cave, e loro coordinate). Non basta limitarsi a riferimenti generici alle tecniche algoritmiche utilizzate (es: “l’operazione X si risolve con un algoritmo greedy”) ma è necessario dettagliare le procedure da utilizzare, tramite pseudocodice o direttamente il linguaggio C. In caso si scelga di consegnare porzioni di codice C, renderle eseguibili e testabili da sole, scrivendo per ognuna di queste porzioni una adeguata funzione `main`.

In particolare si richiede di analizzare, in funzione delle scelte di progettazione fatte, quale risulta essere il costo delle diverse operazioni richieste dalla specifica.

Si richiede inoltre di fornire una rassegna *esauriente* di esempi che potrebbero essere usati per testare il programma e che mettono in evidenza particolari caratteristiche del suo funzionamento (non solo casi tipici di input, ma anche casi limite e/o situazioni patologiche; input che evidenzino la differenza di prestazioni tra le soluzioni progettuali scelte e altre più semplicistiche).

Non si richiede un’implementazione completa del progetto; è sufficiente fornire l’ossatura del programma (eventualmente suddiviso su più file) contenente in particolare: le definizioni dei tipi fondamentali, i prototipi delle funzioni che realizzano le operazioni specificate nella traccia, e tutte le porzioni di codice utili ad illustrarne il loro uso.

Facoltativamente, è possibile consegnare un’implementazione completa e funzionante del progetto.

2.1 Operazioni da eseguire

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

- **territorio**(*nomefile*)

Legge dal file *nomefile* la specifica delle posizioni delle cave presenti in un territorio. Calcola la percorrenza totale minima e un possibile valore dell’ordinata y della tratta del canale che realizza tale percorrenza minima. Se il file *nomefile* non esiste, stampa un messaggio opportuno. È compito dello studente specificare completamente il formato del file.

- **nuova_cava**(x, y)

Apri una nuova cava c nel territorio in posizione $p(c) = (x, y)$. Se esiste già una cava c' tale che $p(c') = (x, y')$, distrugge c' e la sostituisce con c .

- **cava_esaurita**(x, y)

Elimina dal territorio la cava di posizione (x, y) , se tale cava esiste.

- **percorrenza_minima**()

Calcola la percorrenza totale minima rispetto alle cave attualmente attive nel territorio e un possibile valore dell’ordinata y della tratta del canale che realizza tale percorrenza minima.

- **siti_cerimoniali**()

Calcola la posizione dei siti cerimoniali rispetto alle cave attualmente attive nel territorio.

- **sito**(*sito, nomefile*)

Rimuove tutti i blocchi attualmente presenti nel sito *sito* e li rimpiazza con quelli specificati nel sito *nomefile*. Tale file deve specificare la collocazione di ogni blocco. Se il sito non è uno dei due presenti nel territorio, annulla questa operazione. Se il file *nomefile* non esiste, stampa un messaggio opportuno. È compito dello studente specificare completamente il formato del file.

- **blocco**(*sito, x, y, z, x₀, y₀*)
 Inserisce nel sito *sito* un blocco $B(x, y, z)$ in posizione $(x_0, y_0, 0)$. Se la collocazione di tale blocco viene a sovrapporsi a blocchi già presenti nel sito, o se il sito non è uno dei due presenti nel territorio, annulla questa operazione.
- **forza_blocco**(*sito, x, y, z, x₀, y₀*)
 Inserisce nel sito *sito* un blocco $B(x, y, z)$ in posizione $(x_0, y_0, 0)$, rimuovendo nel contempo tutti i blocchi già presenti che vengono a sovrapporsi alla collocazione del nuovo blocco. Se il sito non è uno dei due presenti nel territorio, annulla questa operazione.
- **ruota_blocco**(*sito, x, y, z, direzione*)
 Nel sito *sito*, ruota il blocco che contiene il punto (x, y, z) verso *direzione*, dove *direzione* $\in \{\text{nord, est, sud, ovest}\}$. Se tale punto non appartiene ad alcun blocco del sito, o se il blocco a cui appartiene non può essere liberamente ruotato nella direzione specificata, annulla questa operazione. Se il sito non è uno dei due presenti nel territorio, annulla questa operazione.
- **rimuovi_blocco**(*sito, x, y, z*)
 Elimina dal sito *sito* il blocco che contiene il punto (x, y, z) . Se tale punto non appartiene ad alcun blocco del sito, annulla questa operazione. Se il sito non è uno dei due presenti nel territorio, annulla questa operazione.
- **area**(*sito*)
 Calcola l'area minima di una regione rettangolare contenente le basi di tutti i blocchi presenti nel sito *sito*. Se il sito non è uno dei due presenti nel territorio, annulla questa operazione.
- **stele**(*sito, x₀, y₀, x₁, y₁*)
 Calcola un cammino di lunghezza minima per la stele la cui posizione iniziale è $(x_0, y_0, 0)$ e la cui posizione finale è $(x_1, y_1, 0)$. Il percorso cerimoniale è costituito da tutto il sito cerimoniale, coi blocchi nelle loro posizioni attuali come ostacoli. Stampa il cammino trovato e la sua lunghezza, oppure un opportuno messaggio se tale cammino non esiste. Se il sito non è uno dei due presenti nel territorio, annulla questa operazione.
- **stele_e_percorso**(*sito, nomefile*)
 Legge dal file *nomefile* la posizione iniziale e quella finale della stele, e il percorso cerimoniale entro il quale deve compiersi il cammino. Calcola quindi un cammino di lunghezza minima con questi dati. I blocchi del sito la cui collocazione si sovrappone anche parzialmente al percorso cerimoniale vanno considerati ostacoli. Stampa il cammino trovato e la sua lunghezza, oppure un opportuno messaggio se tale cammino non esiste. Se il sito non è uno dei due presenti nel territorio, annulla questa operazione. Se il file *nomefile* non esiste, stampa un messaggio opportuno. È compito dello studente specificare completamente il formato del file.

3 Modalità di consegna

La presente traccia è valida per l'appello dell'11 settembre 2013.

La relazione (non meno di 3, non più di 10 pagine in formato pdf o rtf) va inviata per posta elettronica all'indirizzo aguzzoli@dsi.unimi.it entro il 18 settembre 2013.

La relazione e gli altri file aggiuntivi (file sorgenti C, esempi di input, ecc) devono essere contenuti in un unico archivio **.zip** il cui nome dovrà essere della forma **cognome.matricola.zip**. Tutti i file nell'archivio, compresa la relazione, devono riportare nome, cognome e matricola dell'autore.

In generale non è prevista una discussione orale dei progetti, ma in casi particolari potranno essere richiesti dei chiarimenti via mail o dal vivo all'autore delle relazioni.

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

La versione aggiornata del progetto è pubblicata in **.pdf** sul sito:

<http://homes.di.unimi.it/~aguzzoli/algo.html>.

Si consiglia di consultare periodicamente questo sito per eventuali correzioni e/o precisazioni relative al testo del progetto.