

Esercitazione 9

- Moli (5.2.10)
- Impianto di risalita (4.2.2)

Un porto ha `num_moli` moli di ugual lunghezza intera `lun`.

A ogni molo si accede da entrambi i lati.

Una nave è caratterizzata da un intero che ne indica lo spazio necessario per attraccare a un molo.

Le navi che chiedono di attraccare sono `n` e lo spazio richiesto da ciascuna nave è un intero contenuto in una cella del vettore `navi`.

Si scriva in C una funzione ricorsiva che faccia attraccare, se possibile, le navi ai moli minimizzando il numero di moli utilizzati.

Modello: partizioni di un insieme (navi) in un numero minimo di blocchi (lati dei moli)

Algoritmo di Er + pruning (si effettua ricorsione su un molo già utilizzato solo se la nave non eccede la capacità residua del molo)

Includiamo anche un controllo di inesistenza della soluzione

- Navi più lunghe dei moli
- Lunghezza complessiva maggiore della capacità dei moli

Un impianto sciistico è composto da diversi skilift ed è utilizzato da sciatori.

Ogni sciatore ha una tessera caratterizzata da un identificatore (cardId) costituito da un numero intero.

Ogni skilift ha un identificatore di skilift (skiliftId) costituito da una stringa alfanumerica di 10 caratteri e un lettore di tessere per abilitare uno sciatore all'utilizzo dello skilift stesso.

Il sistema riceve le letture degli skilift da file

Su ogni riga è scritta la terna skiliftId cardId time.

Ad ogni input il sistema verifica e rilascia o meno l'autorizzazione al passaggio dello sciatore con l'obiettivo di evitare che persone diverse usino la stessa tessera.

L'autorizzazione viene concessa solo se la tessera non è stata letta dallo stesso skilift per un certo intervallo di tempo.

Il tempo è un valore intero rappresentante il numero di minuti trascorsi a partire dalle ore 00:00 del giorno stesso.

Il file contiene righe skiliftId timeInterval (espresso in minuti)

Il sistema deve

- mantenere in opportune strutture dati l'elenco di tutti gli skilift utilizzati da ciascuno sciatore e per ciascuno skilift l'ora dell'ultimo utilizzo (abilitazione)

- fornire una funzione di autorizzazione del tipo

```
int authorize(int cardId, char *skiliftId, int time);
```

che valuti se lo sciatore cardId possa utilizzare lo skilift skiliftId

Essendo il numero di sciatori relativamente elevato e variabile durante la giornata ed essendo i numeri interi usati come card non contigui e potenzialmente grandi, la complessità di tutte le operazioni deve essere al più logaritmica nel numero di sciatori, mentre, essendo il numero di skilift limitato e noto, la complessità delle operazioni non ha alcun vincolo in relazione al numero di skilift.

Struttura dati:

- ADT I class per skilift e sciatori
- Gli sciatori sono memorizzati in una tabella di simboli realizzata tramite BST
- Gli skilift sono memorizzati in una tabella di simboli realizzata tramite vettore non ordinato

Il main, dopo aver opportunamente inizializzato gli skilift, ricerca, per ogni richiesta, il relativo sciatore nella tabella degli sciatori e il corrispondente skilift nella tabella degli skilift.

Se lo sciatore è nuovo, viene inserito, altrimenti viene valutato l'ultimo passaggio e determinata l'eventuale autorizzazione (con eventuale aggiornamento del tempo)