



Esercitazione di laboratorio n. 10

(Caricamento sul portale entro le 23.59 del 23/01/2017 di entrambi gli esercizi)

Esercizio n.1: Impianto di risalita

Si richiede lo sviluppo di un'applicazione in grado di gestire un impianto sciistico secondo le seguenti modalità:

- 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
- ogni skilift ha un lettore di tessere per abilitare uno sciatore all'utilizzo dello skilift stesso.

Il sistema riceve da tastiera le letture degli skilift con formato:

```
skiliftId cardId time
```

Ad ogni input il sistema verifica e rilascia o meno l'autorizzazione al passaggio dello sciatore. Il tempo viene introdotto quale valore intero rappresentante il numero di minuti trascorsi a partire dalle ore 00:00 del giorno stesso. L'obiettivo è evitare che persone diverse usino la stessa tessera. L'autorizzazione viene quindi data solo se la tessera non è stata letta dallo stesso skilift per un certo intervallo di tempo. I tempi dei vari skilift vengono letti da un file, il cui nome viene ricevuto dall'applicazione come parametro sulla linea di comando e ha il seguente formato:

- sulla prima riga appare il numero `S` di skilift
- seguono `S` coppie `skiliftId timeInterval`, dove `timeInterval` è un valore espresso in minuti.

Il sistema, una volta inizializzato, deve:

- mantenere, in memoria centrale, 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 permetta allo sciatore `cardId` di utilizzare lo skilift `skiliftId` oppure gli neghi l'accesso se il vincolo temporale, valutato in base all'ultimo passaggio dello sciatore su tale skilift e l'ora attuale `time`, non è rispettato

Si osservi che:

- *essendo il numero di sciatori relativamente elevato e variabile durante la giornata (e i numeri interi usati come `cardId` non contigui e potenzialmente grandi), il costo asintotico di tutte le operazioni effettuate deve essere al più logaritmico nel numero di sciatori. A tale scopo, si impongono la soluzione appoggiandosi a un BST di sciatori, avente come criterio di ordinamento il loro `cardId`.*
- *si consiglia poi, per ogni sciatore, di mantenere un vettore o una lista degli skilift utilizzati (mantenendo, per ognuno, il tempo di ultimo accesso).*
- *essendo il numero di skilift limitato e di numero noto, il costo asintotico di tutte le operazioni non ha alcun vincolo in relazione al numero di skilift.*



Esercizio n.2: Questioni di inventario

Un avventuriero deve organizzare il proprio inventario per decidere quali oggetti portare con sé nella sua prossima missione. L'inventario è rappresentato da una griglia rettangolare di dimensioni (intere) $inv1 \times inv2$.

Ad ogni oggetto sono associati un nome e una categoria. Ogni oggetto occupa nell'inventario uno spazio rettangolare di dimensioni (intere) $dim1 \times dim2$ e ad ogni oggetto è associata una valutazione legata alla sua stimata utilità (numero intero positivo).

Gli oggetti sono elencati in un file (`equip.txt`), caratterizzato dal seguente formato:

- sulla prima riga è riportato il numero O di oggetti
- seguono O righe nella forma: `nomeOggetto categoria dim1 dim2 utilità`

Si scriva un algoritmo in C che, una volta acquisiti i dati da file e le dimensioni dell'inventario, sia in grado di determinare l'insieme di oggetti da aggiungere all'inventario, e la loro disposizione, tale per cui:

- l'utilità complessiva degli oggetti scelti è massima
- tutti gli oggetti sono posizionabili nell'inventario senza sovrapposizioni
- tutti gli oggetti sono posizionabili nell'inventario senza uscire dai bordi
- deve essere scelto almeno un oggetto di ognuna delle seguenti categorie:
 - arma
 - armatura
 - pozione

Per semplicità, si assuma che non sia possibile ruotare un oggetto.

Nella figura a seguire è proposto un esempio relativo a un inventario 5×6 contenente una spada (5×2), una corazza (3×3), una fiaschetta (2×1), una pila d'oro (2×2) e una piuma (2×1). Nell'inventario rimane ancora una sezione 3×1 completamente libera.

