

## **Gestione di un sistema di ascensori**

Tema d'esame, giugno 2019

### **Introduzione**

L'azienda per cui lavorate ha appena vinto un appalto per la realizzazione di un sistema di ascensori per un hotel di lusso. La struttura ha 20 piani e sono previsti due ascensori della capienza massima di 12 persone ciascuno. Per verificare il dimensionamento degli ascensori, vi è richiesto di implementare un simulatore di utilizzo degli ascensori per effettuare dei test.

Il funzionamento del sistema è il seguente. Ad ogni piano si possono accodare una o più persone, che segnalano la loro presenza premendo il pulsante di chiamata ascensore, e rimangono in attesa finché l'ascensore non arriva al piano – poiché la prenotazione è segnalata da un LED, se un utente arriva e trova altre persone in coda, non è tenuto a premere il pulsante di chiamata di nuovo. Esiste un solo pulsante di chiamata dell'ascensore (non ci sono i due pulsanti per chiamare l'ascensore in salita o in discesa). All'arrivo della cabina al piano, escono tutti i clienti presenti in essa e diretti al piano corrente, dopodiché entrano tutti i clienti in attesa al piano, in ordine di arrivo, fino al raggiungimento della capienza massima. Un cliente che arriva all'ascensore mentre la cabina è al piano è autorizzato ad entrare subito, senza fare alcuna attesa. Entrando in ascensore, ogni persona preme il pulsante corrispondente al piano desiderato (ed è questo il momento in cui il simulatore dell'ascensore conosce quali sono le destinazioni dei clienti). Non è possibile che una persona si metta in coda e poi rinunci a prendere l'ascensore, né che cambi idea sul piano di destinazione.

Il tempo del simulatore è discretizzato, nel senso che esiste una funzione che fa avanzare il riferimento temporale di una unità. Ciascun ascensore è in grado di salire o scendere di un piano all'interno di un'unità di tempo. Il tempo necessario all'ascensore per aprire e chiudere le porte, e alle persone per entrare e uscire dall'ascensore è trascurato: una volta arrivato ad un determinato piano, se ci sono persone in attesa dell'ascensore a quel piano, oppure ci sono clienti in cabina che desiderano scendere, l'ascensore apre le porte e lascia salire e scendere i clienti in un tempo nullo. La politica del movimento degli ascensori, ovvero l'insieme delle regole che determinano il percorso che le cabine compiono, è a vostra scelta. L'efficienza di tale politica rientra nei criteri di valutazione del progetto.

### **Eventi del sistema**

Per quanto detto sopra, il simulatore si compone del sistema di gestione degli ascensori, e di quello di gestione delle code ai vari piani, che sono disgiunti. Esistono i seguenti eventi:

- arrivo di un cliente ad un piano (chiamata dell'ascensore),
- avanzamento di uno slot di tempo.

Il sistema di gestione ascensori che dovete progettare deve decidere come comandare le cabine in modo da rispondere a tali eventi, usando le strutture dati che ritenete più opportune. In particolare, dovete tenere traccia di:

- piani ai quali ci sono clienti in attesa (comunicati al sistema mediante la pressione del pulsante di chiamata),
- piani a cui i clienti dentro la cabina devono andare (comunicati al sistema mediante la pressione del pulsante del piano).

Oltre a ciò, dovete creare anche il sistema di code ai vari piani, che tiene traccia di:

- quanti clienti sono arrivati,

- il piano a cui desiderano andare.

Questo secondo sistema non deve essere collegato a quello di gestione degli ascensori. I due sistemi, però, interagiscono: per esempio, all'arrivo di una cabina a un piano, il sistema che gestisce le code fa in modo che i clienti in attesa entrino nell'ascensore, e che comunichino all'ascensore a che piano sono destinati.

## Inizializzazione e interfaccia

All'inizio del programma, tutte le cabine sono al piano 0, in attesa di utenti. Tutte le code ai vari piani sono vuote. Deve essere possibile interagire con il sistema di simulazione mediante le seguenti funzioni:

<code>add_customer</code>	Arrivo di un cliente ad un determinato piano (con indicazione del piano di destinazione)
<code>time_step</code>	Avanzamento del tempo di un passo
<code>print_system_status</code>	Stampa dello stato del sistema: piano corrente delle cabine, clienti presenti in cabina, piani di destinazione previsti

Potete ovviamente aggiungere ulteriori funzioni, ma quelle elencate in tabella devono essere presenti.

## Input

Il sistema riceve gli eventi via file. Esiste un file di input, passato come argomento da riga di comando, che contiene un evento per ogni riga. Le righe presenti nel file hanno uno dei seguenti formati:

- `PERSON_ARRIVES <piano di arrivo> <piano di destinazione>`
- `TIME_STEP`
- `PRINT_STATUS`

per esempio:

```
PERSON_ARRIVES 0 19
PERSON_ARRIVES 3 0
PERSON_ARRIVES 20 15
TIME_STEP
TIME_STEP
PERSON_ARRIVES 0 20
PRINT_STATUS
TIME_STEP
...
```

Si può dare per scontato che il file abbia un formato corretto e che esista (non è necessario verificare che i comandi scritti siano validi, né che il file esista).

## Modalità di consegna

Il compito deve essere consegnato su elearning, caricando un archivio che include una directory contenente:

- il vostro codice sorgente,

- file di input (almeno quattro) che avete usato per provare il vostro software, scritti secondo i criteri descritti nel paragrafo “Input”,
- un file di testo contenente:
  - riga di comando usata per compilare,
  - qualche esempio di output di esecuzione,

L'archivio non deve contenere l'eseguibile, perché il sorgente sarà compilato in sede di esame. Vi ricordo che, in fase di correzione, il software sarà compilato con le opzioni di ottimizzazione (-O2), per rendere più evidenti eventuali memory leak.

Se non riuscite a gestire la lettura da file di input, nel software devono essere presenti alcune funzioni, chiamate `test_1`, `test_2`, ... (almeno 4) che contengono diverse configurazioni di eventi. Il main contiene chiamate a questa funzione. Questo serve durante la correzione, per provare diverse sequenze di eventi.

## Note allo svolgimento

- Il testo richiede la gestione di un sistema composto da due ascensori. Se è troppo complicato, potete semplificare il problema usando un solo ascensore (**questo comporta una penalizzazione nel punteggio**).
- Il testo richiede la lettura di un file di input. Se non riuscite a implementarla, potete rinunciare e limitarvi a utilizzare l'interfaccia riportata nella tabella sopra (**questo comporta una penalizzazione nel punteggio**).