

Laboratorio di Algoritmi

Progetto “Superette” (novembre 2025)

Nota: La scadenza del progetto è fissata per venerdì 7 novembre **compreso**.

Nota: Si consiglia di consultare sulla pagina web il documento che riporta le avvertenze utili per lo svolgimento del progetto. Si consiglia anche di verificare di tanto in tanto gli aggiornamenti a questo documento, che potranno riportare risposte ai dubbi degli studenti e correzioni di eventuali errori.

Il problema Con il termine “superette” si definisce un punto vendita alimentare al dettaglio con una superficie che non supera poche centinaia di metri. Una superette vuole riorganizzare gli orari di apertura delle proprie casse in modo da trovare un compromesso efficace fra i costi sostenuti e il livello di servizio offerto ai clienti. Per farlo, intende costruire un simulatore del proprio funzionamento, che tenga in considerazione la distribuzione spaziale delle merci, il comportamento dei clienti (ricavato da osservazioni storiche) e diverse ipotesi alternative per gli orari di apertura.

F7	F6	F5	F4	F3	F2	F1	F0
S75	S65	S55	S45	S35	S25	S15	S05
S74	S64	S54	S44	S34	S24	S14	S04
S73	S63	S53	S43	S33	S23	S13	S03
S72	S62	S52	S42	S32	S22	S12	S02
S71	S61	S51	S41	S31	S21	S11	S01
C7	C6	C5	C4	C3	C2	C1	I

Figura 1: Struttura del minimarket

La Figura 1 illustra la struttura della superette, che è piuttosto standard. Si compone di:

- un ingresso, indicato con il simbolo I;
- otto corsie, ciascuna formata da 5 scaffali indicati con sigle progressive: la sigla S_{ij} associata ad ogni scaffale definisce:
 - la corsia con il numero progressivo i , che cresce da 0 a 7 allontanandosi via via dall’ingresso;
 - lo scaffale con il numero progressivo j , che cresce da 1 a 5 procedendo dall’ingresso verso il fondo;

- un corridoio finale, in cui ciascuna corsia sbocca in un punto indicato nella figura con il simbolo Fi , dove i è l'indice della corsia;
- un corridoio iniziale, in cui ciascuna corsia da 1 a 7 sbocca in una cassa, indicata con il simbolo Ci ; il corridoio consente di spostarsi da una cassa all'altra, ma non di raggiungere direttamente la corsia 0;
- un'uscita (non indicata nel disegno), subito dopo le casse.

Per ogni cassa è definito un intervallo di apertura, descritto dalle ore di apertura e di chiusura, misurate in secondi trascorsi dall'apertura del supermercato (la cassa è aperta nei due istanti estremi).

I clienti entrano nel minimarket dall'ingresso e possono muoversi come segue:

- dall'ingresso allo scaffale $S01$ o viceversa, in 5 secondi;
- da uno scaffale al successivo o al precedente nella stessa corsia, in 10 secondi;
- dall'ultimo scaffale di ogni corsia ($Si5$, per i da 0 a 7) allo sbocco della corsia nel corridoio finale (Fi , per i da 0 a 7) o viceversa, in 5 secondi;
- dallo sbocco di una corsia (Fi) a quello precedente o successivo, in 7 secondi;
- dallo primo scaffale di ogni corsia ($Si1$, per i da 1 a 7) alla cassa corrispondente (Ci , per i da 1 a 7) o viceversa, in 5 secondi
- da una cassa (Ci) a quella precedente o successiva, in 8 secondi.

La simulazione del comportamento dei clienti fornisce per ciascuno:

- un'ora di ingresso, espressa in secondi trascorsi a partire dall'apertura del supermercato;
- una lista di merci da acquistare, costituita dal loro numero seguito dai simboli degli scaffali in cui si trovano.

Si suppone che il cliente si muova seguendo esattamente la lista, anche se questo significa girare per il minimarket in modo stupido. Fra i vari percorsi possibili per andare da uno scaffale all'altro, però, il cliente sceglie sempre il più breve. Ogni oggetto richiede 15 secondi per essere raccolto. Raccolto l'ultimo oggetto, il cliente si dirige a una cassa, scegliendo quella raggiungibile nel tempo più breve fra quelle che risulteranno aperte nel momento del suo potenziale arrivo (incluse quelle che stanno per aprire e saranno aperte al suo arrivo ed escluse quelle che sono aperte, ma saranno chiuse al suo arrivo). Se vi sono più casse raggiungibili nello stesso momento, sceglie la più vicina all'ingresso. Questo è ovviamente irrealistico, dato che un vero cliente terrebbe conto della lunghezza della coda alla cassa, ma lo facciamo per semplificare il problema.

Appena arrivato alla cassa, il cliente si mette in coda. Se non ci sono altri clienti, mette tutti i propri acquisti istantaneamente sul nastro, mentre la cassa li smaltisce impiegando 5 secondi per ogni oggetto. Al termine, il cliente paga ed esce istantaneamente.

Il progetto Il simulatore da realizzare consiste in un programma che riceve informazioni da un file di testo. Il file si apre con le informazioni relative agli intervalli di apertura delle 7 casse. Ciascuna delle prime 7 righe si riferisce a una cassa e riporta un codice identificativo alfanumerico di esattamente 5 caratteri¹, l'ora di apertura e l'ora di chiusura (numeri interi). Le casse sono riportate in ordine di distanza crescente dall'ingresso. Per esempio:

```
C0001 0 14400
C0002 0 8000
C0007 7200 14400
C0005 6000 9000
C0003 2000 10000
C0004 6000 9000
C0006 7200 14400
```

indica che le casse C0001 e C0002 aprono all'apertura della superette e chiudono 14400 e 8000 secondi più tardi, rispettivamente. La cassa C0007 apre 7200 secondi dopo l'apertura della superette e chiude 14400 secondi dopo. Considerazioni analoghe valgono per le altre casse. Si ricordi che le casse sono ordinate a partire dall'ingresso e i loro codici alfanumerici non ne indicano la posizione. Per esempio, C0001 è la cassa più vicina, ma C0006 è la più lontana e C0007 la terza in ordine di distanza.

La riga successiva riporta il numero di clienti. Seguono le informazioni relative al loro comportamento. Ogni riga si riferisce a un cliente e ne riporta il cognome (per semplicità, una singola parola di al massimo 15 caratteri²), l'ora di ingresso nel minimarket e il numero di merci da acquistare (numeri interi) e l'elenco dei simboli degli scaffali in cui si trovano, nell'ordine che il cliente seguirà per raccogliarli. L'elenco può contenere oggetti ripetuti, sia in posizioni consecutive (con tempo nullo per il trasferimento fra l'uno e l'altro, ma non nullo per la raccolta) oppure inframmezzati da altri oggetti (col relativo movimento ciclico di va e vieni). Per esempio:

```
4
Utente1 0 4 S13 S21 S45 S52
Utente2 0 1 S51
Utente3 100 2 S32 S25
Utente4 0 2 S25 S21
```

indica che il cliente **Utente1** entra all'apertura del minimarket e compra 4 oggetti, che si trovano, rispettivamente, nel terzo scaffale della prima corsia nel primo della seconda, nel quinto della quarta e nel secondo della quinta. Anche il cliente **Utente2** entra all'apertura della superette e compra un solo oggetto, che si trova nel primo scaffale della quinta corsia. Il cliente **Utente3** entra dopo 100 secondi dall'apertura e compra due oggetti, e così via.

Il programma deve simulare lo svolgimento della giornata gestendone gli eventi significativi. Alcuni eventi sono noti a priori (apertura e chiusura delle casse, ingressi dei clienti), altri sono il risultato delle interazioni fra eventi diversi, che modificano lo stato del sistema, influenzando in tal modo gli eventi successivi. L'idea fondamentale del procedimento è:

¹Qualora non fosse chiaro, "esattamente 5" è un numero costante, come lo sono 7 (casse), 8 (corsie) e 5 (scaffali).

²Al contrario della precedente, questa indicazione serve solo a semplificare il caricamento dei dati. Non va intesa come un'istigazione a ignorare la lunghezza delle stringhe nelle analisi di complessità spaziale e temporale, né tanto meno a considerare costante il numero di possibili diversi clienti.

- l'insieme degli eventi contiene inizialmente quelli noti a priori;
- il primo evento in ordine cronologico viene elaborato per aggiornare lo stato del sistema ed, eventualmente, innescare nuovi eventi futuri:
 - l'ingresso di un cliente consente di dedurre l'ora alla quale il cliente finirà i propri acquisti e arriverà alla cassa scelta, e quindi innesca l'arrivo alla cassa;
 - l'arrivo di un cliente a una cassa innesca l'evento di inizio del servizio:
 1. se la cassa è libera, subito (occupando la cassa per un tempo noto);
 2. se la cassa è occupata, quando i clienti attualmente in coda sono stati tutti serviti e la cassa torna libera;
 - l'inizio del servizio di un cliente innesca la fine del servizio, che coincide con la sua uscita dal minimarket;
 - l'uscita di un cliente dal minimarket lascia immediatamente libera la cassa per il cliente successivo; se non ci sono clienti in coda a quella cassa, essa diventa libera.

I clienti non possono accedere alle casse chiuse. Ogni cliente si accoda nella cassa aperta più vicina alla posizione del suo ultimo acquisto. Dopo l'ora di chiusura, la cassa rimane in servizio per smaltire i clienti accettati in coda, ma non accetta nuovi clienti. I clienti arrivati esattamente all'ora di chiusura sono gli ultimi ad essere accettati. Teoricamente, con queste regole un cliente potrebbe restare prigioniero, ma assumeremo che i dati lo escludano e ogni cliente possa sempre trovare almeno una cassa aperta al termine dei suoi acquisti.

In caso di eventi simultanei, le aperture delle casse precedono sempre gli ingressi dei clienti. Questi precedono le uscite, che precedono gli arrivi in cassa. Gli arrivi precedono gli inizi dei servizi, e questi precedono le chiusure delle casse. Fra eventi dello stesso tipo vale l'ordine alfabetico rispetto agli identificativi dei clienti associati. Per le aperture e chiusure delle casse, che non riguardano alcun cliente, vale l'ordine alfabetico rispetto agli identificativi delle casse associate agli eventi.

Il programma deve per prima cosa stampare tutti gli eventi in ordine cronologico. Ogni evento richiede una riga ed è descritto dall'ora in secondi trascorsi dall'apertura del supermercato (numero intero), dal tipo di evento (**APERTURA**, **CHIUSURA**, **INGRESSO**, **CASSA**, **SERVIZIO**, **USCITA**), dall'identificativo del cliente interessato (per gli eventi di ingresso, arrivo in cassa, servizio e uscita) e infine da quello della cassa interessata (per gli eventi di apertura, chiusura, arrivo in cassa e servizio). Nell'esempio considerato:

```
0 APERTURA C0001
0 APERTURA C0002
0 INGRESSO Utente1
0 INGRESSO Utente2
0 INGRESSO Utente4
100 INGRESSO Utente3
144 CASSA Utente4 C0002
144 SERVIZIO Utente4 C0002
154 USCITA Utente4
174 CASSA Utente2 C0002
174 SERVIZIO Utente2 C0002
179 USCITA Utente2
328 CASSA Utente3 C0002
328 SERVIZIO Utente3 C0002
```

```

330 CASSA Utente1 C0002
338 USCITA Utente3
338 SERVIZIO Utente1 C0002
358 USCITA Utente1
2000 APERTURA C0003
6000 APERTURA C0004
6000 APERTURA C0005
7200 APERTURA C0006
7200 APERTURA C0007
8000 CHIUSURA C0002
9000 CHIUSURA C0004
9000 CHIUSURA C0005
10000 CHIUSURA C0003
14400 CHIUSURA C0001
14400 CHIUSURA C0006
14400 CHIUSURA C0007

```

cioè all'apertura del minimarket aprono le casse C0001 e C0002 ed entrano i clienti nell'ordine: Utente1, Utente2 e Utente4. Quindi entra il cliente Utente3 al secondo 100. Al secondo 144, il cliente U4 arriva alla cassa C2. Siccome questa è aperta e libera, comincia subito il servizio e lo termina al secondo 154. Al secondo 174, il cliente Utente2 arriva alla cassa C0002, che è la più vicina cassa aperta (C0005 sarebbe più vicina al suo ultimo acquisto, ma è chiusa). Siccome la cassa è libera, il servizio comincia subito e termina al secondo 179. Il cliente Utente3 arriva al secondo 328, quindi il cliente Utente1 arriva al secondo 330. È chiaro che nel mondo reale andrebbe alla cassa C0001, ma l'ipotesi è che invece scelga la cassa aperta più vicina, quindi si mette in coda alla cassa C0002. Il cliente Utente3 esce al secondo 338 e subito comincia il servizio di Utente1, che esce al secondo 358. Infine, aprono le casse C0003 (al secondo 2 000), C0004 e C0005 (al secondo 6 000), C0006 e C0007 (al secondo 7 200) e chiudono via via le casse nell'ordine: C2 (al secondo 8 000), C4 e C5 (al secondo 9 000), C3 (al secondo 10 000), C1 14400, C6 e C7 (al secondo 14 400).

Il programma stampa poi le informazioni relative alle casse, riportate in ordine alfabetico rispetto ai loro identificativi. Ogni riga riguarda una cassa e riporta l'identificativo, il numero di secondi durante i quali è stata occupata (numero intero) e la percentuale di tale durata rispetto all'intervallo di apertura (numero reale con due cifre decimali). Questa percentuale supera il 100% nel caso (improbabile) di una cassa occupata durante l'intero intervallo e rimasta aperta per smaltire clienti dopo la chiusura. Nell'esempio riportato più sopra:

```

C0001 0 0.00%
C0002 45 0.56%
C0003 0 0.00%
C0004 0 0.00%
C0005 0 0.00%
C0006 0 0.00%
C0007 0 0.00%

```

dato che solo la cassa C2 è stata occupata, per lo 0.56% del suo tempo di apertura, che corrisponde a 45 secondi su 8 000.

Infine, il programma stampa le informazioni relative ai clienti, sempre in ordine alfabetico. Ogni riga riguarda un cliente e riporta l'identificativo, il numero di secondi di permanenza nel minimarket, il numero di clienti in coda all'arrivo in cassa (compreso quello che sta venendo servito: la coda è vuota solo per clienti che arrivano e vengono serviti immediatamente) e il numero di secondi di permanenza in coda. I tre numeri sono tutti interi. Per esempio:

Utente1 358 1 8
Utente2 179 0 0
Utente3 238 0 0
Utente4 154 0 0

dato che tutti i clienti dell'esempio vengono serviti al loro arrivo in cassa, tranne **Utente1**, che arriva mentre la cassa serve **Utente3** e aspetta 8 secondi.

Dubbi e chiarimenti