



Esercitazione 5 (10-29)



5.1 Selezione da file

- Leggere riga per riga un file di testo
- Riscrivere in un file di output le stesse righe
- Ma, di ciascuna riga, scrivere solo il testo compreso tra parentesi quadre

Segnare in un `bool` se si è letta una parentesi aperta, ma non ancora una parentesi chiusa



5.2 Veicoli ed elicotteri

- Definire una classe astratta per i *veicoli* d'intervento
 - Metodo `estimate_time(x, y)` per il calcolo del tempo d'intervento
- Ridefinire l'*ambulanza* (es. 3.2) come sottoclasse di *veicolo*
 - Richiede **1 minuto** di preparazione alla partenza
 - Viaggia a **90 Km/h**, per la distanza di Manhattan
- Definire una classe *elicottero* come sottoclasse di *veicolo*
 - Richiede **5 minuti** di preparazione
 - Viaggia in linea d'aria a **250 Km/h**



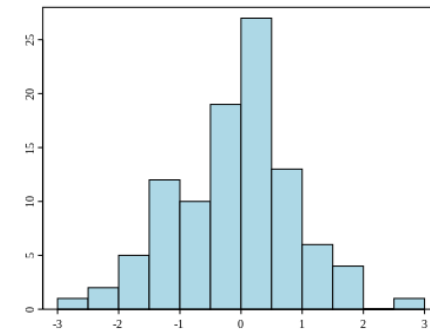
5.3 Lista di veicoli

- Leggere da un file i dati di una lista di *veicoli* (*elicotteri* o *ambulanze*, es. 5.2)
 - In ogni riga ci sono i dati completi di un veicolo, compreso il tipo
- Poi, in un ciclo, chiedere all'utente il punto in cui intervenire
- Il sistema sceglie il veicolo più rapido per l'intervento
- Comunica all'utente:
 - Il nome del veicolo
 - La stima del tempo di intervento



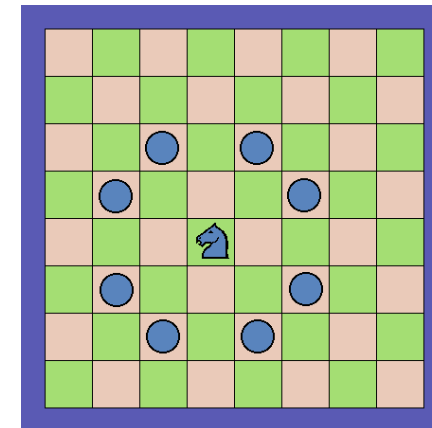
5.4 Istogramma con barre verticali

- Leggere da un file una sequenza di valori (positivi)
- Mostrare un istogramma, in un canvas 600x600
 - Altezza di ciascuna barra proporzionale al valore corrispondente
 - La barra più alta occupa tutto lo spazio disponibile
 - Barre in blu, per valori sotto alla media
 - Barre in rosso, per valori sopra (o uguali) alla media
 - La larghezza del canvas è divisa equamente per le barre



5.5 Percorso del cavallo

- L'utente sceglie dim. scacchiera (≥ 5)
- Allocare matrice quadrata di interi (tutti 0)
- In un ciclo, chiedere all'utente x ed y
 - Segnare nella cella il numero di mosse fatte
 - Si può giocare solo nelle celle vuote, ad "L" dall'ultima giocata
 - Dopo ogni mossa, stampare la matrice aggiornata
 - Controllare se la matrice è piena



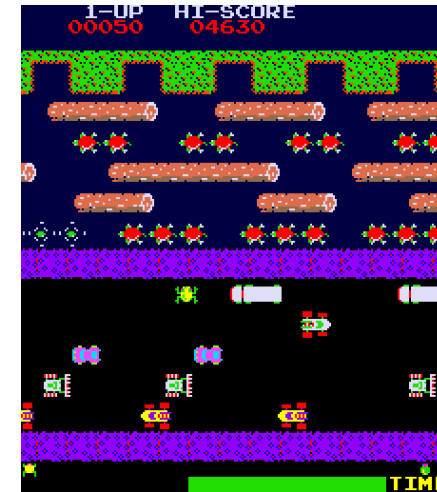
```
1  14  9  20  3
24 19  2  15 10
13  8 25  4 21
18 23  6 11 16
7  12 17 22  5
```

OUTPUT



5.6 Frogger, zattere

- Modificare la classe `Ball` dell'esempio `bounce`
 - Il movimento è solo orizzontale (destra o sinistra)
 - Quando un oggetto esce da un bordo, riappare dal bordo opposto
 - Dare però al movimento uno spazio più ampio dell'arena visibile
- Inserire la classe nel gioco, come `Raft`
- La rana viene trascinata, quando poggia su una zattera
 - Controllare che il centro della rana sia dentro la zattera
 - Quando la rana collide con una zattera, può memorizzarla in un campo
 - Metodo *getter* che ottenere la velocità di una zattera
 - La rana muore, se trascinata fuori dallo schermo



5.7 Frogger, fiume

- Aggiungere il fiume al gioco Frogger
 - Personaggio fisso
 - Disposto sotto alle zattere (prima, nell'arena)
- Se la rana urta il fiume (ma non una zattera), muore
- Se la rana attraversa il fiume, vince

