

Esercizio 1 – 25 punti

Con l'avvicinarsi delle vacanze estive, si vuole realizzare in Python un programma per gestire la lista delle cose da mettere in valigia. Il programma, organizzato a menù, deve consentire di:

- 1) aggiungere un nuovo oggetto alla lista
- 2) eliminare uno specifico oggetto dalla lista
- 3) visualizzare l'elenco degli oggetti presenti nella lista
- 4) salvare l'elenco su file
- 5) caricare da file un elenco già fatto

Per il punto 4 occorre chiedere all'utente il nome del file nel quale salvare la lista. Il file di testo deve avere estensione “.myl”

Per il punto 5 il programma deve prima mostrare l'elenco di tutti i file “.myl” presenti nella cartella attuale e poi chiedere all'utente di indicare il nome (o il progressivo) del file dal quale si vuole caricare la lista. Una scelta non valida deve portare a chiedere nuovamente tale informazione.

Il caricamento da file cancella sempre tutto ciò che è già presente in memoria.

Oltre a sviluppare il software richiesto, organizzandolo in funzioni, lo studente implementi i dovuti controlli sui dati e si assicuri che le eccezioni vengano gestite correttamente.

Esercizio 2 – 20 punti senza bonus, 25 punti con il bonus

Si vuole realizzare un programma Python che esegue la conversione tra le seguenti valute:

Valuta	Sigla internazionale
Euro	EUR
Dollaro americano	USD
Franco svizzero	CHF
Dollaro inglese	GBP
Dollaro australiano	AUD
Yen giapponese	JPY

L'utente sceglie la valuta di partenza e la valuta di arrivo; inserisce poi l'importo e ottiene il corrispettivo.

Il tasso di cambio è variabile e dipende dalla giornata. Per eseguire la conversione si vuole utilizzare il tasso di cambio della giornata nella quale l'utente usa il programma (giornata odierna).

Per ricavare il tasso di cambio utilizzare la seguente API

`http://api.opencambio.org/v1/currencies/{valuta di partenza}/rates?start={data_odierna}&end={data_odierna}`

Ad esempio per sapere quanto vale il dollaro oggi rispetto alle varie valute (porre attenzione al formato della data):

`http://api.opencambio.org/v1/currencies/USD/rates?start=2024-05-22&end=2024-05-22`

Oltre a sviluppare il software richiesto, organizzandolo in funzioni, lo studente implementi i dovuti controlli sui dati e si assicuri che le eccezioni vengano gestite correttamente.

Se serve, la documentazione della API la trovate qui: <https://github.com/lerocha/open-cambio>

Per fare i test è consentito usare convertitori online.

Bonus: realizzare il programma implementando una interfaccia basata su finestra grafica anziché su console

Esercizio 3 – 35 punti senza bonus, 40 punti con il bonus



Si vuole realizzare in Python un gioco simile a “La ruota della fortuna”.

Il gioco viene disputato tra vari giocatori, a turno, e consiste nell’indovinare una frase che viene “scoperta” poco per volta, man mano che i giocatori richiedono le lettere.

Pima di iniziare la partita il programma carica le frasi misteriose ed il relativo titolo di aiuto (file frasi.txt).

```
Andare al mare in estate  
Vacanze  
Una mela al giorno toglie il medico di turno  
Proverbio
```

Quindi chiede quanti giocatori vogliono giocare, i relativi nomi (o nickname) ed un colore per contraddistinguere il giocatore. Ad ogni giocatore viene assegnato un montepremi iniziale di 0€. Infine, viene selezionata in maniera automatica e del tutto casuale una tra le frasi misteriose. A quel punto inizia la partita.

A turno, viene visualizzato il nome del giocatore che deve giocare, nome colorato secondo il colore scelto all’inizio. Viene anche visualizzato il suo montepremi attuale e la frase parzialmente coperta (ad. esempio con degli “*” o “-”).

Il giocatore deve indicare cosa vuole fare:

- girare la ruota
 - quando gira la ruota viene estratta in modo casuale e visualizzata una delle voci in figura



- se la voce selezionata è un premio in denaro, il giocatore può richiedere una consonante (non possono essere richieste vocali); se la consonante è presente nella frase misteriosa, allora il suo montepremi viene aggiornato (premio * numero consonanti presenti nella frase) e viene mostrata la frase misteriosa aggiornata; se la consonante non è presente, il montepremi rimane invariato. Non è possibile richiedere consonanti già richieste in precedenza;
- se la voce selezionata è “passa-mano” il turno passa immediatamente al giocatore successivo;
- se la voce selezionata è “perde-tutto” il turno passa immediatamente al giocatore successivo ed inoltre il montepremi del giocatore viene azzerato;
- se la voce selezionata è “jolly”, il giocatore acquisisce un jolly, che potrà usare dal turno successivo;

- usare un jolly (se ne ha almeno uno) → non gira la ruota, compra subito una vocale gratuitamente. Se la vocale è presente nella frase misteriosa, allora il suo montepremi viene aggiornato (premio * numero vocali presenti nella frase) e viene mostrata la frase misteriosa aggiornata; se la vocale non è presente, il montepremi rimane invariato. Non è possibile richiedere vocali già richieste in precedenza;
- comprare una vocale: l'acquisto dimezza il montepremi del giocatore.

A meno che dalla ruota non sia uscito “passa-mano” o “perdi-tutto”, prima di passare il turno, al giocatore viene chiesto se desidera dare la soluzione. Una soluzione corretta raddoppia il montepremi del giocatore e pone fine alla partita. Una soluzione errata non provoca alcun effetto, se non quello di passare il turno al giocatore successivo.

Alla fine della partita occorre stampare una classifica dei giocatori, da quello che ha il montepremi più alto a quello che ha il montepremi più basso.

Oltre a sviluppare il software richiesto, organizzandolo in funzioni, lo studente implementi i dovuti controlli sui dati e si assicuri che le eccezioni vengano gestite correttamente.

Bonus

I montepremi di tutti i giocatori vanno salvati su un file “montepremi.txt”. Il file deve essere strutturato in maniera tale da fare in modo che ogni volta che si avvia il gioco viene mostrata la classifica di tutti i giocatori che hanno giocato con il relativo montepremi totale ed il numero di partite giocate da sempre. La classifica va ordinata per montepremi, dal più alto al più basso.

Esempio:

- al primo avvio la classifica sarà vuota;
- giocano Alessio, Aldo e Maria. Al termine della partita i punteggi sono Alessio:10000 , Aldo: 15000, Maria: 2000
- all'avvio successivo la classifica mostrerà Aldo: 15000 (1) Alessio:10000 (1) Maria: 2000 (1). Questa volta giocano Alessio, Maria e Sergio totalizzando questi punti Alessio: 1000, Maria: 1000, Sergio: 1000
- all'avvio successivo la classifica mostrerà Aldo: 15000 (1) Alessio:11000 (2) Maria: 3000 (2) Sergio: 1000 (1).

Modalità di consegna

Esclusivamente su piattaforma GITHUB.

Tabella dei punteggi

Voce	Punti base	Punti bonus
Esercizio 1	25	
Esercizio 2	20	25
Esercizio 3	35	40
Totale	80	85
Qualità codice suddivisione in funzioni, commenti al codice, strutture dati utilizzate, implementazione algoritmi	10	
Totale	90	100