Compito di Programmazione
9 Febbraio 2024

N.B. Ogni esercizio deve essere svolto su un foglio diverso. Scrivere Nome, Cognome e Matricola su ogni foglio.

1. **(punti 8)** Scrivere la seguente funzione `bilanciata` che prende in input un array di 1000 caratteri "(", ")", "[", "]", "{", "}" oppure " "(spazio bianco) e verifica se la stringa in input è bilanciata (ovvero per ogni tipo di parentesi aperta vi sia una corrispondente parentesi chiusa e l'ultima parentesi aperta è la prima ad essere chiusa con una parentesi dello stesso tipo, e così via). Ad esempio, sequenze del tipo "([)]" non sono bilanciate, mentre sequenze come "([ ])[{ } ( )]" sono bilanciate.

   La funzione può avere in input altri parametri, come ad esempio *altri array*, e si possono usare funzioni ausiliarie purchè definite. [Max punti 6 se qualche funzione utilizza l'iterazione.]

2. **(punti 8)** L'Ikea sta aggiornando i manuali con le istruzioni per costruire i mobili. Un manuale è una lista di istruzioni per la costruzione di un singolo mobile. Ogni istruzione ha i seguenti campi: una descrizione e i pezzi necessari per quell'istruzione (es. vite, asse, ecc.). La descrizione viene memorizzata come array di char e i pezzi si memorizzano con un array di bool. Se il pezzo in posizione i ha valore true allora viene utilizzato per quell'istruzione.
   Per esempio *[true, false, false, true]* indica che per l'istruzione vengono usati i pezzi con indice 0 e 3.

   Si richiede di implementare la struttura dati necessaria per rappresentare la lista delle istruzioni e le seguenti funzioni:

   - `remove_istruzioni` che prende in input una lista di istruzioni `L` e una descrizione `d` e ritorna la lista `L` in cui ogni elemento con descrizione `d` è stato rimosso (ci possono essere più descrizioni uguali in `L`).
   - `occorrenze_pezzo` che prende in input la lista delle istruzioni e la posizione di un pezzo nell'array (int) e restituisce un intero che indica il numero di volte che quel pezzo viene utilizzato per tutta la lista di istruzioni.
   - `aggiungi_pezzo` che prende in input una lista di istruzioni `L`, una descrizione `d` e la posizione `i` di un pezzo e ritorna la lista `L` modificata in cui l'istruzione corrispondente a `d` ha anche il pezzo `i`.

   [**N.B.** Non è possibile utilizzare le liste bidirezionali.]
   [**N.B.** È possibile utilizzare le funzioni `strncpy`, `strcpy` e `strcmp` della libreria `cstring`.]
   [**N.B.** Non è possibile restituire puntatori a `char`.]
   [**N.B.** Si possono usare funzioni ausiliarie se definite.]

3. **(punti 8)** Una banca emette una carta prepagata sulla quale un utente può effettuare versamenti. La carta è caratterizzata dal budget a disposizione e può essere utilizzata solo per pagamenti online. Si implementi la classe `CartaPrepagata`, il relativo costruttore e i metodi:

   - `versamento()`: il quale riceve la cifra da versare e incrementa di conseguenza il budget della carta.
   - `pagamento_online()`: il quale riceve la cifra del pagamento, controlla che il budget sia sufficiente e in caso positivo sottrae la cifra. La funzione deve restituire l'esito dell'operazione.

   La stessa banca emette un altro tipo di carta prepagata che prevede anche una funzione di prelievo di contanti. Questa nuova carta è caratterizzata **anche** da un secondo valore (detto deposito) utilizzato solo per i prelievi, da un costo di prelievo e da un valore percentuale. I prelievi possono però essere effettuati solo dal deposito e non dal budget, quest'ultimo è riservato solo alle operazioni online. Il valore percentuale invece viene utilizzato per calcolare quanto di ogni versamento deve essere destinato al deposito e quanto al budget della carta.
   Si implementi la classe `CartaDeposito`, il relativo costruttore e si modifichi, sfruttando opportunamente l'ereditarietà, il metodo `versamento()`. Inoltre, si implementi il metodo `prelievo()`, il quale riceve la cifra da prelevare, controlla che il deposito sia sufficiente e in caso positivo scala la cifra e il costo di prelievo. La funzione deve restituire l'esito dell'operazione.