

Una macchina per la distribuzione di bevande e frutta "fresca" è programmata in C#.

La macchina acquisisce solo monete ed è in grado di dare il resto.

Si deve realizzare un programma WinForm che simuli il sistema di pagamento della macchina.

Per la gestione dei pagamenti, le classi usate dal software devono essere organizzate così:

Gestione delle monete:

E' definita una Classe base "Moneta", dalla quale sono derivate le classi: "m\_5cent", "m\_10cent", "m\_20cent", "m\_50cent", "m\_1e", "m\_2e" (realizzarne almeno 3).

La classe "Moneta" deve avere almeno questi attributi:

- nessuno (aggiungerne se ne servono!)

Le classi derivate devono avere almeno questi attributi:

Proprietà

- Diametro, proprietà **statica** a virgola mobile, di 32 bit.
- Massa, proprietà **statica** a virgola mobile, di 32 bit.

Le due proprietà devono essere di sola lettura e vengono assegnate nel costruttore di default (quello **senza** parametri).

Classe Distributore (derivata da Form, nella simulazione)

Contiene il programma principale e gli oggetti che realizzano il programma; contiene anche i metodi generali, cioè quelli che non è il caso di incorporare in una classe.

Questa classe deve contenere almeno:

- Una **struttura dati** che contenga tutte le monete attualmente presenti in macchina (array, ArrayList o SortedList, a scelta) (il programma deve sapere quante e quali monete sono dentro la macchina).
- Una **struttura dati** (tre array, un array di struct o un'Hashtable, a scelta) che contenga tutti i prodotti presenti in macchina, memorizzandone la descrizione (codice univoco che ha solo questo prodotto), la quantità presente in macchina ed il prezzo.
- Un bottone "NuovaMoneta" con relativo evento click, che simula l'inserzione di una moneta da parte dell'utente (per la simulazione: Diametro e Massa della moneta inserita possono essere letti da una TextBox).  
Se la moneta non è accettabile, nella simulazione viene aperta una MessageBox di avvertimento e non si crea nessuna istanza di classe Moneta.  
Se la moneta è accettabile:
  - Si deve creare un oggetto di quelli derivati da Moneta ("m\_XXX"), la cui classe corrisponde alla moneta che il sistema di pagamento ha riconosciuto.
  - Si deve mettere l'oggetto creato nella struttura dati che ricorda le monete che sono dentro alla macchina.
  - Si deve calcolare e visualizzare il credito in denaro accumulato dall'utente.
- Un bottone "SceltaProdotto", con relativo evento click, che simula la scelta di un prodotto (la descrizione potrà essere presa da una TextBox).  
Alla partenza di questo evento si dovrà:
  - chiamare un metodo "Eroga" (da **non** scrivere) che restituisce un valore di tipo bool, true se il prodotto è stato effettivamente dato al cliente (se non viene dato, il sistema prova per tre volte, se non ci riesce restituisce le monete di credito)
  - erogare il resto, visualizzando (in qualche modo) le **monete** di resto che vengono date all'utente (**ogni** moneta che viene data) e cancellarle dalla struttura dati che tiene traccia delle monete (si deve sapere che la macchina **non ha più** quelle monete).
  - Se la macchina non ha più monete per dare il resto deve farlo sapere e lasciare un credito del valore delle monete di resto che non riesce a restituire.
  - La macchina dovrà dare un allarme se finisce un tipo di prodotto.
- Un bottone "RidammiISoldi", il cui evento click fa restituire al cliente le monete relative al credito attuale.

Aggiungere alle classi i metodi e le proprietà che si ritengono opportune, in modo da completare le funzionalità richieste dal programma.

**Nota:** per decidere quali monete restituire, fa comodo che la struttura dati delle monete sia ordinata. Se non si è in grado di realizzare l'ordinamento, affidarsi ad un metodo "sort()", che, "magicamente", ordini la struttura.