PROVA 21/09/21

Si vuole realizzare un applicazione per la gestione degli incassi di un parco giochi. Il parco ospita diverse attrazioni , alcune delle quali sono direttamente dipendenti dal parco , altre sono gestite da terze parti . Per accedere alle attrazioni che dipendono dal parco , i clienti acquistano una scheda precaricata che al momento dell’accesso al parco si puo’ supporre che abbia un valore X>=M, (M è il valore minimo) mentre per accedere alle attrazioni “indipendenti” si deve pagare in contanti . Ogni volta che un cliente accede ad una attrazione dipendente dal parco , il credito della scheda viene decrementato del relativo costo del biglietto . Non è possibile spendere piu’ di quanto c’è sulla scheda.

La classe scheda ha le variabili membro codice (oggetto di tipo string) che identifica univocamente una scheda , una costante M che rappresenta il valore minimo (numero intero), una variabile membro X (intero) che rappresenta il credito iniziale della scheda e una variabile membro Corrente (intero) che rappresenta il credito residuo e alla quale viene dato valore X. Quando una scheda viene costruita deve essere X>=M, in caso contrario a X viene dato il valore M. La classe scheda fornisce le funzioni getter, e una funzione in ingresso riceve il costo del biglietto di una attrazione e ogni volta che viene chiamata decrementa Corrente di tale valore . Se il credito residuo sulla scheda non è sufficiente viene lanciata un eccezione e Corrente non viene decrementata.

Si realizzi quindi una classe astratta “attrazione” che implementa il comportamento generale di una attrazione del parco : ha come variabili membro un nome , ad esempio “casa degli spiriti” (sia una stringa C allocata dinamicamente) , un costo (numero reale) . La classe fornisce le funzioni getter e setter, una funzione che consente di acquistatre un biglietto e una funzione che consente di calcolare l’incasso realizzato. Tali funzioni sono implementate differentemente se l’attrazione è gestita dal parco o se è indipendente , come specificato di seguito.  
La classe che realizza il comportamento di un attrazione gestita dal parco aggiunge una variabile membro che la mette in relazione con un numero variabile di schede. Sia tale variabile membro un vettore allocato dinamicamente, la relazione con la classe scheda deve essere individuata e coerentemente realizzata dallo studente. La classe deve consentire l’inserimento nel vettore senza duplicati. L’inserimento viene effettuato nella prima posizione libera ma se il codice della scheda è già presente non viene effettuato, la funzione che serve ad acquistare il biglietto riceve in ingresso il codice della scheda e modifica il valore corrente della scheda corrispondente a tale codice utilizzando la funzione membro “decrementa” della classe scheda. Si realizzi anche la funzione che calcola e restituisce l’incasso complessivo dell’attrazione.

La classe che realizza il comportamento “indipendente” aggiunge il nome del titolare dell’attrazione (stringa C allocata dinamicamente) e una variabile membro che serve a registrare l’incasso corrente, la funzione che serve ad acquistare il biglietto riceve in ingresso una stringa che deve essere confrontata con il nome del titolare, se il confronto va a buon fine (le stringhe sono uguali) modifica il valore corrente dell’incasso aggiungendovi il costo dell’attrazione. In caso contrario l’acquisto non puo’ essere fatto. La funzione che calcola l’incasso complessivo dell’attrazione semplicemente restituisce il valore della variabile membro che memorizzi l’incasso corrente.

Le classi devono fornire le funzionalità necessario per una corretta gestione dell’estensione dinamica ove presente.  
Si realizzi un opportuno programma di test delle classi e delle funzionalità realizzate, che realizza il parco giochi attraverso un opportuno vettore C di attrazioni e dopo aver effettuato acquisti sugli(?) elementi del vettore calcola e stampa a video l’incasso totale effettuato