

FONDAMENTI DI INFORMATICA I
FOND. DI INFORMATICA E PROGRAMMAZIONE A OGGETTI

Una *Prepagata* è una carta di debito usa e getta sulla quale viene pre-caricato un credito iniziale all'atto dell'emissione. Successivamente queste carte non possono più essere ricaricate: possono essere utilizzate unicamente per effettuare acquisti (**in numero potenzialmente illimitato**). Sono ammessi anche eventuali acquisti allo scoperto, ossia in assenza di credito residuo sufficiente.



Su una carta *Prepagata* deve essere possibile effettuare le seguenti operazioni:

PRIMA PARTE (*qualora siano presenti errori di compilazione, collegamento o esecuzione in questa parte, l'intera prova verrà considerata insufficiente e pertanto non verrà corretta*)

✓ **Prepagata p1 (crIn) ;**

Costruttore che crea una carta con credito iniziale pari all'intero `crIn`. All'inizio la lista degli acquisti è vuota.

✓ **p1.acquista (imp, luogo) ;**

Operazione che aggiunge l'acquisto di importo pari all'intero `imp` alla lista degli acquisti effettuati con la carta (`luogo` è una stringa **di lunghezza arbitraria** che indica l'esercizio commerciale presso il quale è stato effettuato l'acquisto) e restituisce `true`. Se `imp < 0` l'acquisto non viene aggiunto alla lista e la funzione restituisce `false`. L'acquisto viene invece aggiunto alla lista anche se il saldo corrente non lo permetterebbe (acquisto *Scoperto*).

✓ **cout<<p1 ;**

Operatore di uscita per il tipo *Prepagata*. Un esempio di uscita è il seguente:

```
40<C>[90=>50]Esselunga
35<C>[50=>15]Carrefour
22<S>[15=>-7]Ikea
```

dove viene mostrata un *Prepagata* con credito iniziale pari a 90 euro, con la quale sono stati effettuati tre acquisti, di 40, 35 e 22 euro (in questo esatto ordine cronologico) all'*Esselunga*, al *Carrefour* e all'*Ikea*, rispettivamente (fra parentesi quadre vengono riportati i saldi parziali prima e dopo l'acquisto, separati da un uguale seguito da un maggiore). Se l'acquisto era scoperto viene stampata una *S* maiuscola fra parentesi angolari, altrimenti viene stampata una *C* maiuscola (acquisto *Coperto*).

NB: non vanno aggiunti spazi fra i vari campi.

SECONDA PARTE (*si invita a mettere sotto commento le operazioni di questa seconda parte che dovessero impedire la compilazione, il collegamento o la corretta esecuzione del codice*)

✓ **int (p1) ;**

Operatore di conversione di una *Prepagata* ad intero. Restituisce il saldo corrente, considerando anche gli eventuali acquisti scoperti (pertanto il saldo *può* essere negativo).

✓ **!p1 ;**

Operatore di negazione logica che modifica la carta `p1` eliminando da essa tutti gli eventuali acquisti effettuati allo scoperto.

✓ **Prepagata p2 (p1) ;**

Costruttore di copia che inizializza la carta `p2` con il valore della carta `p1`.

✓ **~Prepagata () ;**

Distruttore.

Mediante il Linguaggio C++, realizzare il tipo di dato astratto *Prepagata*, definito dalle precedenti specifiche. **Individuare le eventuali situazioni di errore e metterne in opera un corretto trattamento.**

L'uscita che ci si aspetta dal programma è la seguente (si noti l'assenza di righe completamente vuote)

```
-----
Test del costruttore, della acquista e dell'operatore di uscita
(deve stampare: 40<C>[90=>50]Esselunga )
40<C>[90=>50]Esselunga
-----
Ora deve stampare anche questa riga: '35<C>[50=>15]Carrefour'
40<C>[90=>50]Esselunga
35<C>[50=>15]Carrefour
-----
Ed ora anche questa riga: '22<S>[15=>-7]Ikea'
40<C>[90=>50]Esselunga
35<C>[50=>15]Carrefour
22<S>[15=>-7]Ikea
-----
Visualizzo il saldo potenziale della carta p1 (deve stampare -7)
-7
-----
Test dell'operatore ! (debbono essere presente solo acq. coperti)
40<C>[90=>50]Esselunga
35<C>[50=>15]Carrefour
-----
Test del costruttore di copia (il cont. di p2 deve essere = a p1)
40<C>[90=>50]Esselunga
35<C>[50=>15]Carrefour
-----
Test (implicito) del distruttore
-----
```

NOTE SULLO SVOLGIMENTO DELLA PROVA PRATICA

AVVIO E IDENTIFICAZIONE

- Avviare la macchina in modalità diskless, scegliere “Fondamenti di Informatica I” ed effettuare il login:
nome: studenti
password: studenti
- Aprire un terminale e spostarsi sulla cartella ‘elaborato’ (\$ cd ~/elaborato). Si utilizzi il comando pwd per verificare che ci si trovi nella cartella corretta /home/studenti/elaborato.
- Dare il comando \$ ident, sempre da dentro la cartella. Lo script richiede i propri dati (cognome, nome, numero di matricola e password (la password **non va dimenticata** in quanto è indispensabile per scaricare da internet il proprio elaborato a consegna avvenuta). Il comando ident crea il file *matricola.txt* nella cartella corrente. Lo script può essere lanciato più volte, in tal caso il file *matricola.txt* viene sovrascritto. Per verificare che il file sia stato creato e che il contenuto sia quello giusto dare il comando (la password è codificata):
\$ cat /home/studenti/elaborato/matricola.txt
- A questo punto il docente verifica che tutti gli studenti abbiamo effettuato l'identificazione, dopodiché provvede a inviare i seguenti file nella cartella elaborato del proprio PC: *compito.h*, *compito.cpp*, *main.cpp*.
Controllare pertanto che questi file, insieme al file *matricola.txt*, siano presenti sul proprio elaboratore.

SVOLGIMENTO DELLA PROVA

- Definire ed implementare il tipo di dato astratto richiesto e le relative funzioni nei file *compito.h* e *compito.cpp*. Il file *main.cpp* contiene la funzione principale *main()* ed è utilizzato dallo studente per testare la sua implementazione della classe. Il file *main.cpp* può essere modificato a piacere. In sede di valutazione dell'elaborato verrà considerato **esclusivamente il contenuto dei file compito.h e compito.cpp** ed è pertanto **vietato cambiare nome a tali file**.

Per compilare e linkare dare il comando:

```
$ g++ main.cpp compito.cpp (eseguibile invocabile tramite $ ./a.out)
(utilizzare g++ -g per includere le informazioni di debug qualora si intenda debuggare con ddd).
```

PER CONSEGNARE O RITIRARSI

Recarsi dal docente avendo preso nota dell'identificativo della macchina (g34, s23, ...).