

Nome:

Matricola:

Università degli Studi di Napoli “Federico II”
P1 9CFU - Prova pratica del 31 gennaio 2017 -TURNO2
Prof. Valeria Vittorini

[illegible]

*Lo studente legga attentamente il testo e produca il programma ed i casi di test necessari per dimostrarne il funzionamento. La mancata compilazione dell'elaborato o la compilazione con errori daranno luogo alla valutazione come **prova non superata**.*

[illegible]

Testo della Prova

Si realizzi la gerarchia *cliente - cliente registrato*. La classe *cliente* ha le variabili membro *codice cliente* (stringa C allocata staticamente) *profilo* (ad esempio, *administratore*, *user* etc... stringa C allocata dinamicamente) . .

La classe cliente registrato estende la base aggiungendo le variabile membro *login*, *password* (stringhe C allocate dinamicamente) e *ora di registrazione* (di tipo Timestamp). La classe Timestamp è realizzata dallo studente e fornisce una semplice rappresentazione di marcatura oraria mediante tre variabili membro di tipo intero rappresentante ora, minuti e secondi. Si forniscano per la classe Timestamp gli operatori di shift e le funzioni di utilità get e set. Non sono richiesti particolari controlli sulla validità del timestamp.

Entrambe le classi della gerarchia forniscono una funzione membro per il controllo di validità dei dati della classe. La funzione fornita dalla classe base controlla che il codice utente sia una stringa di almeno 7 caratteri alfanumerici, la funzione della classe derivata ridefinisce quella base aggiungendo al controllo sul codice cliente anche il controllo sulla password, che deve essere di almeno 8 caratteri e contenere almeno una cifra e almeno una lettera alfabetica maiuscola. Le funzioni per il controllo di validità generano una eccezione nel caso in cui il controllo fallisca. Si implementi una classe eccezione che oltre al consueto messaggio di errore fornisca anche una informazione relativa a quale controllo è fallito (codice cliente o password non validi). Si forniscano per entrambe le classi le funzioni necessarie ad una corretta gestione dell'estensione dinamica, l'overloading degli operatori di shift e degli operatori relazionali == e !=. Il programma utente memorizza in un vettore una sequenza di riferimenti ad oggetti sia di tipo base che di tipo derivato, e stampa su file di tipo testo solo i dati relativi ad oggetti che superano il relativo controllo di validità. Attenzione! Le password non devono essere stampate in chiaro sul file!!!! Vengono sostituite da tanti * quanti sono i caratteri che le compongono.

Si realizzi infine un opportuno programma di test delle classi realizzate.