

**Nome:**

**Matricola:**

**Università degli Studi di Napoli “Federico II”**  
**P1 9CFU - Prova pratica del 31 gennaio 2017 -TURNO1**  
**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 *utente - utente di posta elettronica*. La classe *utente* ha le variabili membro *codice fiscale* (stringa C allocata staticamente) *cognome* (stringa C allocata dinamicamente) e *data di registrazione* (di tipo Data).

La classe Data è realizzata dallo studente e fornisce una semplice rappresentazione di una data mediante tre variabili membro di tipo intero. Si forniscano per la classe data gli operatori di shift e le funzioni di utilità get e set. Non sono richiesti particolari controlli sulla validità della data.

La classe utente di posta elettronica estende la base aggiungendo la variabile membro *indirizzo\_email* (stringa allocata dinamicamente). Entrambe le classi forniscono una funzione membro per il controllo di validità dei dati della classe. La funzione fornita dalla classe base controlla che il codice fiscale sia una stringa di 16 caratteri alfanumerici, la funzione della classe derivata ridefinisce quella base aggiungendo al controllo sul codice fiscale anche il controllo sulla variabile membro *indirizzo*, che deve contenere il carattere @ (uno e uno solo). 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 fiscale o indirizzo errati). 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.

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