

**Corso Programmazione ad Oggetti**  
**Corso di Laurea in Informatica**  
**Esercitazione di Laboratorio del 08/04/2016**

**ESERCIZIO 1** Utilizzare la classe *Contatto*, allegata alla traccia, per realizzare una classe *GestoreContatti*.

Dotare la classe di un dato private che sia una lista di contatti, e dei seguenti metodi che permettano di gestire la lista:

- **void stampaContatti ()**, che stampi su standard output il contenuto della lista.
- **bool inserisciContatto (const Contatto& c)**, che aggiunga il contatto *c*, alla fine della lista, se non esiste già un contatto uguale. Restituire true se è stato inserito, false altrimenti.
- **int numeroContatti ()**, che restituisca il numero di contatti presenti nella lista.
- **void ordinaContatti ()**, che ordini la lista di contatti.
- **bool eliminaContattoConRemove (const Contatto& c)**, che elimini dalla lista il contatto *c*, se presente. Utilizzare a questo scopo il metodo *remove* di *list* (<http://www.cplusplus.com/reference/list/list/remove/>). Restituire true se è stato eliminato, false altrimenti.
- **bool eliminaContattoConErase (const Contatto& c)**, che come il metodo precedente, elimini dalla lista il contatto *c*, se presente. Tuttavia, in questo caso, si utilizzi il metodo *erase* di *list*. Si faccia attenzione al fatto che il metodo *erase* invalida gli iteratori che fanno riferimento agli elementi rimossi (<http://www.cplusplus.com/reference/list/list/erase/>). Restituire true se è stato eliminato, false altrimenti.
- **vector<string> trovaNumeri (const string& nome, const string& cognome)**, che restituisca un vettore contenente i numeri di telefono memorizzati nella lista in corrispondenza dei contatti con il nome ed il cognome ricevuti come parametri.
- **string trovaCognomePiuFrequente ()**, che restituisca una stringa corrispondente al cognome con il maggior numero di occorrenze nella lista. Se la lista è vuota restituire una stringa vuota.
- **bool verificaDueContattiStessoTelefono ()**, che restituisca true se nella lista sono presenti almeno due contatti con nome e/o cognome diversi, ma stesso numero di telefono, false altrimenti.

Realizzare un main in cui viene creato un oggetto di tipo *GestoreContatti*, e viene controllata la correttezza dei metodi implementati. Verificare, inoltre, che l'effetto dei due metodi per l'eliminazione sia lo stesso.

**ESERCIZIO 2** Realizzare una classe *template Tripla*. Dotare la classe di alcuni costruttori e dei metodi *get* e *set*. Dotare inoltre la classe degli operatori di inserimento su stream (operator<<), di estrazione su stream (operator>>), e degli operatori relazionali == e >. Realizzare un main in cui si creano triple di diverso tipo e si verifichi la correttezza dei metodi implementati.

**Alcuni suggerimenti per lo svolgimento:**

- Non utilizzare caratteri accentati per nomi di campi, variabili o metodi.
- Per il carattere tilde (~) occorre digitare: ALT + 126 su sistemi Windows, ALT gr + ` su sistemi Linux, ALT + 5 su sistemi Mac OS X.
- Per ogni esercizio realizzare una cartella contenente tutti i file creati.
- Si consiglia di utilizzare un semplice editor di testo (ad esempio, gedit), e compilare da linea di comando. Per compilare occorre passare tutti i file sorgente (.cpp) al compilatore.
- Minimizzare le inclusioni.