UNIVERSITÀ DELLA CALABRIA

Facoltà di Scienze Matematiche Fisiche e Naturali
Corso di Laurea in Informatica
Prova di Programmazione ad Oggetti
Data: 1 aprile 2007

Esercizio 1

Siano date la classe *Nuotatore* e la classe *Torneo* aventi le seguenti intestazioni:

```
enum Ruolo { STILE = 0, RANA,
             DELFINO, TUFFI};
class Nuotatore
{
private:
    string nome;
    string cognome;
    Ruolo r;
public:
    Ruolo ottieniSpecialita();
    string ottieniNome();
    string ottieniCognome();
    bool operator==(const
                    Nuotatore&)
...
}
```

```
class Campionato
{
private:
    string nome;
    list<Nuotatore> giocatori;
public:
    string ottieniNome();
    list<Nuotatore>& classificaFinale(Ruolo);
}
```

Queste classi fanno parte di un sistema che gestisce i dati storici dei campionati di nuoto. Supponendo che tutto quello di cui non è richiesta l'implementazione (classi suddette e corrispondenti classi lista) siano già implementate, si realizzino le seguenti due funzioni:

```
1)  string Nuotatore::ottieniNome();
2)  list<Nuotatore> haVintoInAlmenoDueSpecialita(const Campionato& C);
```

Più in dettaglio: la funzione (1) restituisce il contenuto del membro *nome* della classe *Nuotatore*; la funzione (2) restituisce la lista dei nuotatori che hanno vinto in almeno due specialità nel campionato in input.
Nota: E' possibile aggiungere, qualora lo studente lo ritenga necessario, ulteriori membri pubblici o privati alle classi. Nel caso in cui venga aggiunta una nuova funzione, deve esserne fornita l'implementazione. Si considerino implementati tutti i metodi già presenti nella traccia.

Esercizio 2

Realizzare l'interfaccia della classe *PilaPerPrimaIVincitori* che implementa una pila di priorità di Nuotatori in cui i nuotatori che entrano sono serviti in base al numero di vittorie e, a parità di vittorie, nel consueto ordine (LIFO). Implementare, inoltre, almeno il metodo push() della pila.

Esercizio 3

Una multinazionale del tabacco ha commissionato un software per la gestione dei punti vendita affiliati e dei vari prodotti. Il sistema deve essere in grado di effettuare delle statistiche sulla vendita dei prodotti e sui venditori tra cui: (a) determinare il numero di prodotti venduti di un certo tipo, (b) determinare il prodotto più venduto, (c) ordinare i prodotti in base alle vendite (d) costruire un elenco ordinato di venditori in base alle vendite effettuate, (e) determinare il prodotto e il punto vendita con la resa maggiore (maggiori ricavi)
Progettare le classi che consentono di rispondere alla necessità del committente. **E' richiesta la realizzazione dei soli file di intestazione.**

Nota: è possibile utilizzare le librerie standard STL.