

# Progetto Programmazione 2, a.a. 2007/2008

PROF. FRANCESCO RANZATO

## 1 Scopo

Lo scopo del progetto è lo sviluppo in C++ di un sistema minimale per la gestione di un (modesto) database tramite una interfaccia utente grafica.

## 2 Template di classe Contenitore<K>

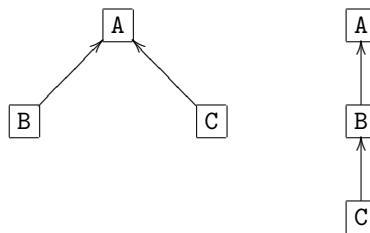
Definire un template di classe `Contenitore<K>` i cui oggetti rappresentano un contenitore di oggetti di tipo `K`. Il template di classe `Contenitore<K>` deve fornire almeno delle funzionalità minime di inserimento, rimozione e ricerca.

Vi è un unico vincolo da rispettare nello sviluppo del template di classe `Contenitore<K>`, **non** è permesso l'utilizzo dei contenitori della libreria STL.

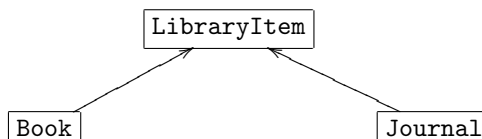
Si scelga l'implementazione di `Contenitore<K>` ritenuta più opportuna per usare il contenitore come struttura dati per il database.

## 3 Gerarchia di classi

Definire una gerarchia di classi  $G$  composta almeno da tre classi `A`, `B` e `C` che modellano una realtà di oggetti da gestire tramite il database. Le gerarchie possibili per le tre classi `A`, `B` e `C` sono le seguenti:



Ad esempio, nel codice scheletro che integra il presente documento, la gerarchia di classi è la seguente:



Le classi della gerarchia dovranno essere dotate di opportune e realistiche interfacce pubbliche.

Usare la fantasia ed ispirarsi ai propri interessi personali nello scegliere la realtà da modellare tramite questa gerarchia di classi. Saranno accettate anche gerarchie più complesse che contengono più di tre classi.

## 4 Interfaccia Grafica

Si richiede di sviluppare una semplice interfaccia grafica (GUI, Graphic User Interface) usando la libreria QT (ultima versione a Novembre 2007: **versione 4.3**, per Linux/Windows/MacOS, home page: [trolltech.com/products/qt](http://trolltech.com/products/qt)) che permetta all'utente di gestire da interfaccia il database di oggetti delle classi della gerarchia  $G$ . Si tratta quindi di sviluppare una GUI per la gestione di un contenitore  $C$

di puntatori polimorfi alla superclasse  $A$  di ogni altra classe della gerarchia  $G$ , che come requisito minimale permetta inserimenti, rimozioni e ricerche di oggetti della gerarchia  $G$  nel database rappresentato dal contenitore  $C$ , ed interrogazioni del database basate sulle funzionalità pubbliche delle classi in  $G$ .

Lo scheletro di codice della GUI che accompagna e complementa il presente documento illustra l'impostazione di base di una semplice GUI e fornisce una intelaiatura minimale per lo sviluppo della GUI. La libreria QT è dotata di una documentazione completa e precisa che sarà la principale guida di riferimento nello sviluppo della GUI. La libreria QT offre una moltitudine di classi e metodi per lo sviluppo di GUI curate, dettagliate, accattivanti e "user-friendly". Sarà apprezzato l'uso di funzionalità di QT diverse da quelle minimali usate nello scheletro della GUI.

## 5 Valutazione del Progetto

Un buon progetto dovrà essere sviluppato seguendo i principi basilari della programmazione orientata agli oggetti, anche per quanto concerne l'interfaccia grafica. La valutazione del progetto prenderà in considerazione i seguenti criteri (in ordine alfabetico):

- Correttezza del progetto (deve compilare e funzionare correttamente)
- Orientazione agli oggetti, modularità ed estensibilità del progetto
- Qualità ed usabilità della GUI
- Quantità e qualità delle funzionalità del progetto (inclusa la gestione della memoria)

Si ricorda inoltre che all'eventuale esame orale lo studente dovrà saper motivare le scelte progettuali e dovrà dimostrare la piena conoscenza di ogni parte del progetto.

## 6 Regole

Il presente documento va inteso come una "specificazione minimale" di progetto, ossia tutto ciò che non è espressamente richiesto è lasciato a scelta libera. Il progetto dovrà essere realizzato da ogni singolo studente in modo **indipendente** da terze persone.

**Relazione:** Il progetto dovrà essere accompagnato da una breve relazione scritta che descriva **sinteticamente** le principali scelte progettuali. La relazione deve essere redatta in **puro formato testo** in un file di nome (preciso) `relazione.txt`. La relazione deve anche specificare il sistema operativo di sviluppo e le versioni precise del compilatore e della libreria QT.

**Compilatore e libreria QT:** Il progetto deve compilare (e quindi eseguire) correttamente sulle macchine **Linux** del laboratorio informatico del plesso Paolotti con il compilatore GNU `g++ 4.x`. Nelle macchine Linux del laboratorio è installata la libreria QT nell'ultima versione 4.3. È naturalmente possibile sviluppare il progetto su altri sistemi operativi come MacOS/Windows. In tal caso, prima di consegnare il progetto, ricordarsi di effettuare (anche remotamente tramite `ssh -X`) una prova di compilazione, esecuzione e funzionamento sulle macchine Linux del laboratorio.

**Cosa consegnare:** tutti i file sorgente `.h` e `.cpp` ed il file `relazione.txt` contenente la relazione. Se la compilazione del progetto necessita di un project file (`.pro`) per `qmake` diverso da quello ottenibile tramite l'invocazione di `qmake -project` allora deve anche essere consegnato un file `progetto.pro` che permetta la generazione automatica tramite `qmake` del `Makefile`.

**Cosa non consegnare:** codice oggetto, eseguibile, file di back-up generati automaticamente da editor (ad esempio i file con ultimo carattere "~" generati da Emacs) e tutto quanto non necessario per la compilazione.

**Come consegnare:** dalle macchine del laboratorio invocando il comando

consegna progetto-programmazione2

dalla directory contenente i file da consegnare. **Non** saranno accettate altre modalità di consegna (ad esempio, via email). Naturalmente è possibile consegnare remotamente il progetto tramite il server

ssh.studenti.math.unipd.it

e opportuni comandi/programmi come ssh, sftp, scp, etc.

**Scadenze di consegna:** Il progetto dovrà essere consegnato rispettando **tassativamente** le scadenze **ufficiali** (data e ora) previste che verranno rese note tramite le liste elettroniche di iscrizione (SIS) agli esami scritti ed orali. Approssimativamente la scadenza sarà circa 7-10 giorni prima dell'esame orale. Per i primi due appelli orali le scadenze saranno le seguenti:

- Esame orale del 19/12/2007  $\Rightarrow$  scadenza: 12/12/2007 ore 23:59.
- Esame orale del 10/1/2008  $\Rightarrow$  scadenza: 2/1/2008 ore 23:59.

Per i progetti ritenuti insufficienti, lo studente dovrà consegnare una nuova versione del progetto.

**Assistenza:** per domande e questioni relative al progetto rivolgersi agli assistenti per il progetto di laboratorio:

dott. Alberto Conz: <albertoconz@gmail.com>

dott. Francesco Tapparo: <tapparo@math.unipd.it>

via email, eventualmente per fissare un incontro.