

qCharts

Progetto di Programmazione ad Oggetti, a.a. 2012/2013

prof. Francesco Ranzato

1 Scopo

Lo scopo del progetto qCharts è lo sviluppo in C++/Qt di un sistema per creare, archiviare, modificare e visualizzare *charts*, ovvero diagrammi di dati.

Un chart è una rappresentazione grafica di una collezione di dati tipicamente numerici, in cui i dati sono rappresentati mediante simboli: ad esempio, come rettangoli in un diagramma a barre (detto anche istogramma), linee in un diagramma a linee, settori di un cerchio in un diagramma a torta. Si rimanda alla voce Chart di Wikipedia/English per una descrizione dettagliata dei charts e per un ampio campionario delle tipologie più comuni di charts.

La specifica **dettagliata** di quali funzionalità il progetto qCharts debba fornire è lasciata in massima parte a libera scelta dello studente, che può trarre ispirazione dal web, dagli innumerevoli charting software esistenti, dai propri interessi e dalla fantasia. La specifica **minimale** delle funzionalità richieste a qCharts consiste in:

1. Creazione e modifica dei dati numerici di un chart
2. Archiviazione (e recupero) su file dei dati numerici di un chart
3. Visualizzazione grafica di un chart con possibilità di scegliere tra almeno tre tipologie diverse di chart

2 Interfaccia Grafica

Si richiede di sviluppare una GUI usando la libreria Qt che permetta all'utente di usufruire agevolmente delle funzionalità di qCharts mediante tale GUI. In particolare, la GUI dovrà permettere la visualizzazione delle varie tipologie di chart rese disponibili da qCharts. È possibile progettare alcune opzioni di qCharts che permettano di specializzare i chart per alcune situazioni di interesse. Alcuni semplici esempi di specializzazione (attuali in questo periodo) potrebbero essere: andamento spread/borse/monete, sondaggi di gradimento di partiti/personaggi politici, classifiche di eventi sportivi, market shares di prodotti come PC/cellulari/tablet, etc. (si faccia ricorso alla fantasia ed ai propri interessi).

Il campione di codice della GUI che accompagna e complementa il presente documento fornisce una intelaiatura minimale a fini esemplificativi per lo sviluppo della GUI. Come noto, la libreria Qt è dotata di una documentazione completa e precisa che sarà la principale guida di riferimento nello sviluppo della GUI, oltre ad offrire i tool QtCreator e QtDesigner. La libreria Qt offre una moltitudine di classi e metodi per lo sviluppo di GUI curate, dettagliate e user-friendly. Sarà apprezzato l'uso di funzionalità di Qt diverse da quelle minimali/standard.

3 Valutazione del Progetto

Un buon progetto dovrà essere sviluppato seguendo i principi fondamentali della programmazione orientata agli oggetti, anche per quanto concerne lo sviluppo dell'interfaccia grafica. La valutazione del progetto prenderà in considerazione i seguenti criteri:

1. **Correttezza:** il progetto deve compilare e funzionare correttamente, e raggiungere correttamente e pienamente gli scopi previsti.

2. **Orientazione agli oggetti:** (A) progettazione ad oggetti, (B) modularità (in particolare, massima separazione tra il codice logico del progetto ed il codice della GUI del progetto), (C) estensibilità e (D) qualità del codice sviluppato.
3. **Quantità e qualità:** quante e quali funzionalità il progetto rende disponibili, e la loro qualità.
4. **GUI:** utilizzo corretto della libreria Qt, qualità ed usabilità della GUI.

4 Esame Orale e Registrazione Voto

La partecipazione all'esame orale è possibile solo dopo:

1. avere superato con successo (almeno 18/30) l'esame scritto
2. avere consegnato il progetto entro la scadenza stabilita
3. essersi iscritti alla lista Uniweb dell'esame orale

Il giorno dell'esame orale (nel luogo ed all'orario stabiliti) verrà comunicato l'esito della valutazione del progetto. Tre esiti saranno possibili:

- (A) Valutazione positiva del progetto con registrazione del voto complessivo proposto **con esenzione dell'esame orale**. Nel caso in cui il voto proposto non sia ritenuto soddisfacente dallo studente, sarà possibile richiedere l'esame orale, che potrà portare a variazioni in positivo o negativo del voto proposto.
- (B) Valutazione del progetto da completarsi con un **esame orale obbligatorio**. Al termine dell'esame orale, o verrà proposto un voto complessivo sufficiente oppure si dovrà riconsegnare il progetto per un successivo esame orale.
- (C) Valutazione negativa del progetto che comporta quindi la **riconsegna del progetto** per un successivo esame orale (il voto dell'esame scritto rimane valido).

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

5 Regole

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

5.1 Relazione

Il progetto dovrà essere obbligatoriamente accompagnato da una **breve** (massimo 6 pagine in formato 10pt) relazione scritta che descriva **sinteticamente** le scelte progettuali ritenute più significative. La relazione deve essere presentata come un file PDF di nome (preciso) `relazione.pdf`. La relazione deve anche specificare il sistema operativo di sviluppo e le versioni precise del compilatore e della libreria Qt.

5.2 Compilatore e libreria Qt

Il progetto deve compilare ed eseguire correttamente sulle macchine **Linux** del laboratorio informatico del plesso Paolotti o della torre Archimede con il compilatore GNU g++ 4.x. Nelle macchine Linux del laboratorio è installata la libreria Qt nella versione 4.6 (5.0 è invece l'ultima versione di Qt del dicembre 2012, mentre l'ultima versione di Qt4 è la 4.8 del settembre 2012). 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.

5.3 Cosa consegnare

Tutti i file sorgente `.h` e `.cpp`, il file `relazione.pdf` contenente la relazione, eventuali file che memorizzano dati necessari per il corretto funzionamento del programma. 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 o IDE e tutto quanto non necessario per la corretta compilazione ed esecuzione del programma.

5.4 Come consegnare

Dalle macchine del laboratorio invocando il comando

```
consegna progetto-pao-2013
```

dalla directory contenente **tutti e soli** 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.

5.5 Scadenze di consegna

Il progetto dovrà essere consegnato rispettando **tassativamente** le scadenze **ufficiali** (data e ora) previste che verranno rese note tramite le liste Uniweb di iscrizione agli esami scritti ed orali. Approssimativamente la scadenza sarà circa 9-10 giorni prima dell'esame orale.

Per i progetti ritenuti insufficienti, lo studente dovrà consegnare una nuova versione del progetto per un successivo appello orale.