



UNIVERSITÀ
DEGLI STUDI
FIRENZE



Esercitazione

Git exercise

Obiettivo

- Il progetto CLion fornito contiene un progetto di un software per implementare un visualizzatore/editor di immagini (operazioni solo simulate)
- Scopo dell'esercitazione è:
 - Esercitarsi nell'uso di Git per versionamento di file
 - Creazione di tag e branch
 - Merge di branch

Schema del codice

- Il programma è composto da 3 file:
- `Main.cpp` che mostra le funzioni fornite dalla classe che implementa la gestione delle immagini bitmap.
- `Bitmap.h` header con dichiarazione dei metodi della classe omonima
- `Bitmap.cpp` implementazione della classe.
- Ci sono anche due file `test.jpg` e `test.png` che emulano due immagini in questi formati

Attività 1

- Partire dal branch master, in cui è impostato lo HEAD del versionamento
 - Aggiungere il file `version.h` in cui includere la definizione:
 - `static const std::string VERSION = "1.0";`
 - Includere `version.h` in `main.cpp` e far stampare il numero di versione.
 - Compilare, correggere eventuali errori e fare il commit della versione 1.0 del programma.
 - Dal menu VCS -> Git -> Tag inserire il tag v1.0 per marcare il fatto che lo stato del versionamento è importante e corrispondente al rilascio del prodotto.

Attività 2

- Simulare lo sviluppo di un branch di bug fix della version 1.0, mentre sul branch master continuerà lo sviluppo di nuove versioni del programma:
 - Creare branch `bugfix_1.0` (VCS -> Git -> Branches -> New Branch, oppure click su nome di branch su barra di stato in basso a destra -> New Branch)
 - Correggere il metodo `resize()` per fare in modo che le dimensioni di larghezza e altezza vengano modificate solo se >0
 - Fare commit
 - Correggere il metodo `load()` per fare in modo che si possano caricare anche i file con estensione `.jpeg`:

```
if (fileExt == "jpg" || fileExt == "jpeg" || fileExt == "png")
```

- Fare commit



Attività 3

- Simulare lo sviluppo del branch master dove continua lo sviluppo della nuova versione del programma:
 - Fare il checkout del branch master (da VCS -> Git -> Branches -> New Branch, oppure click su nome di branch su barra di stato in basso a destra -> selezionare master -> checkout)
 - Aggiungere nel metodo load per fare in modo che si possano caricare anche i file con estensione .bmp

```
if (fileExt == "jpg" || fileExt == "bmp" || fileExt == "png")
```

- Portare la versione del programma a 1.1 in version.h
- Fare commit



Attività 4

- Fare il merge tra il codice del branch di bug fix e il branch master, così da avere il metodo `resize()` corretto ed il metodo `load` sia corretto per leggere `.jpeg` che espanso per leggere `.bmp`:
 - Controllare di essere su branch master e nel caso farne il checkout (da VCS -> Git -> Branches -> New Branch, oppure click su nome di branch su barra di stato in basso a destra -> selezionare master -> checkout)
 - Selezionare il menu popup di branch (da barra di stato o da VCS -> Git -> Branches)
 - Fare il merge
 - Notare come la fusione di `version.h` non dia problemi mentre la fusione di `Bitmap.cpp` in questo caso dia un conflitto dato che sia nello sviluppo del branch master che in quello di bug fix sia toccata la stessa riga dell'`if` che controlla l'estensione del file
 - Risolvere manualmente il merge consentendo al programma di gestire sia le estensioni `.jpeg` (del bugfix) che `.bmp` (del master). Una volta corretto il codice nel pannello centrale della finestra di fusione premere `Accept`.