

Relazione del progetto di Programmazione ad Oggetti: Qontainer

Elisabetta Piombin 1142189

Indice

1	Abstract	1
2	Descrizione della gerarchia di classi	1

1 Abstract

Qontainer è un progetto realizzato al fine di fornire un contenitore che gestisca una libreria di contenuti multimediali: file audio e file video, che si dividono a loro volta in canzoni, podcast, serie tv e film.

Per farlo, la classe templatizzata **container** a sua volta fa uso di altre classi annidate:

1. **nodo**: inserita nella parte privata di **container**, viene usata per memorizzare i vari contenuti multimediali, visti come se fossero una lista concatenata di elementi, con ogni nodo diviso nel suo campo **info** (di tipo parametrico **T** e **next** (di tipo **nodo***). In **container** è presente un puntatore al primo elemento della lista.
2. **const_iterator**: inserita nella parte pubblica di **container**, è la classe che permette l'implementazione di iteratori costanti.
3. **iterator**: inserita nella parte pubblica di **container**, è la classe che permette l'implementazione di iteratori non costanti.

Vista l'assenza di puntatori ad altre classi nella gerarchia, è stata ritenuta superflua l'implementazione di una classe per un eventuale *smart pointer*.

2 Descrizione della gerarchia di classi

La classe base astratta da cui deriva tutta la gerarchia è **contenutomultimediale**, che verrà poi concretizzata tramite sue classi derivate. I metodi virtuali sono **riproduci()**, **pausa()**, **operator==** e il relativo distruttore; **riproduci()** e **pausa()** sono anche puri, poiché la loro implementazione è assegnata alle classi derivate.

Da **contenutomultimediale** derivano immediatamente altre due classi: **audio** e **video**, che sono le due macrocategorie di appartenenza dei file che vengono memorizzati nella libreria.

Da **video** derivano due classi: **film** e **episodio**, mentre da **audio** derivano altre due classi, **podcast** e **canzone**.

Non si verifica la situazione di ereditarietà multipla.