

# AlgaT

Progetto di Algoritmi e Strutture dati  
A.A. 2018-2019

## Indice

<b>ALGAT</b>	<b>1</b>
CHE COS'È	2
INTERFACCIA UTENTE	
<b>PROGETTO</b>	<b>4</b>
STRUTTURA	5
SCELTE IMPLEMENTATIVE	6
<b>REDBLACKTREE</b>	<b>4</b>
CARATTERISTICHE	5
ALGORITMO DI RICERCA	6

# AlgaT

## Che cos'è

AlgaT(ALGA as Tutorial) è un'applicazione stand-alone basata su JavaFX. L'applicazione mostra un tutorial su un argomento a scelta del corso di Algoritmi e Strutture Dati. Questo progetto si presta a mostrare gli alberi bilanciati di ricerca tramite la struttura dati degli alberi RedBlack.

## Interfaccia utente

L'applicazione all'apertura mostra una pagina iniziale contenente un'introduzione sull'argomento scelto e la possibilità di accedere, tramite appositi pulsanti, ai tutorial. I tutorial simulano in modo interattivo il funzionamento della struttura dati degli alberi bilanciati di ricerca. L'applicazione offre due tutorial: il primo tutorial mostra la ricerca, dato in input il valore di una chiave, di un nodo all'interno dell'albero, il secondo mostra le operazioni di inserimento e cancellazione di un nodo, dato in input il valore di una chiave. Il funzionamento di tale struttura dati viene mostrato graficamente e accompagnato da una spiegazione testuale, il tutto avviene tramite un'esecuzione step-by-step gestita dall'utente tramite i relativi bottoni.

Ogni tutorial inoltre permette di accedere ad una fase di domande di autoapprendimento sempre inerenti all'argomento scelto; una volta terminate tutte le domande con esito positivo, l'utente viene riportato alla pagina iniziale.

# Progetto

## Struttura

```
AlgaT_mod
|_____src
|           |_____algaT_mod
|                   |_____AlgaT_mod.java
|                   |_____DomandeController.java
|                   |_____GraficaAlbero.java
|                   |_____PaginaInizialeController.java
|                   |_____TutorialController.java
|                   |_____domande
|                   |           |_____DomandeTutorial1
|                   |           |_____DomandeTutorial2
|                   |
|                   |_____fxml
|                   |           |_____Domande.fxml
|                   |           |_____PaginaIniziale.fxml
|                   |           |_____Tutorial.fxml
|                   |
|                   |_____img
|                   |           |_____RedBlackTree.png
|                   |
|                   |_____tree
|                   |           |_____Nodo.java
|                   |           |_____RedBlackTree.java
```

## Scelte implementative

### Domande di autoapprendimento

Le domande per la fase di autoapprendimento sono memorizzate su file e vengono caricate tramite l'istanziamento di un oggetto `FileReader()` a cui viene passato come parametro il percorso del file da caricare, e un oggetto `BufferedReader()` che ci permette tramite il metodo `readLine()` di leggere riga per riga il file di testo.

Questa scelta rende più facile l'inserimento e/o la modifica delle domande.

Il file di testo che viene caricato è strutturato in modo che la prima riga contenga la domanda e la riga successiva contenga la risposta relativa, e così via.

## Interfaccia grafica

L'interfaccia grafica è stata implementata con l'utilizzo di JavaFX Scene Builder, generando un file con estensione .fxml. E' stata creata un'interfaccia con un design responsive, capace di adattarsi alle diverse dimensioni degli schermi, a tal proposito, le dimensioni dell'applicazione, in fase di apertura, vengono calcolate in proporzione allo schermo del dispositivo su cui viene lanciata.

Ogni pagina fa riferimento ad un controller, una classe java, che gestisce l'inizializzazione della pagina stessa e gli eventi possibili sulle varie componenti della pagina (es. evento di click su un bottone).

Il caricamento delle pagine fxml avviene tramite l'istanziamento di un oggetto FXMLLoader a cui viene passato il percorso del file con estensione .fxml, che poi verrà caricato tramite il metodo load().

## Esecuzione step-by-step

# RedBlackTree

## Caratteristiche

//TO DO

## Algoritmo di ricerca

//TO DO