

# PROGETTO ALGAT

---

Angelo Di Iorio

Università di Bologna

# Progetto AlgaT

- AlgaT(ALGA as Tutorial) è un'applicazione standalone basata su JavaFX
- L'applicazione mostra un tutorial su un argomento a scelta del corso
- La discussione consiste nel mostrare l'esecuzione di AlgaT, usarlo e commentare il codice

# Regole

- Il corso prevede lo svolgimento di un **progetto di gruppo**. I gruppi sono composti di 2-3 persone. **Non sono ammessi gruppi di 4 o più persone.**
- Il progetto va presentato al docente e consegnato due giorni prima della discussione.
- Si consegna una volta sola e il voto è valido per l'intero anno accademico (e per gli anni successivi se le regole non cambiano)
- E' obbligatorio consegnare il progetto prima dell'esame scritto
- Il progetto vale  $\frac{1}{4}$  del voto complessivo (media pesata)
- Il progetto di quest'anno si può presentare fino a Febbraio 2020

# Regole: consegna

- Il progetto si discute di persona (**tutti i membri del gruppo insieme**) e si consegna due giorni prima della discussione
  - mail a: [angelo.diiorio@unibo.it](mailto:angelo.diiorio@unibo.it)
  - link a spazio condiviso (Google Drive, Dropbox o altri spazi)
- La consegna finale deve includere:
  - Sorgenti Java
  - Eseguitibile, Runnable .jar
  - Relazione (max 8 pagine) su struttura dell'applicazione e README

# Discussione progetti

- Date per discutere progetti a giugno/luglio:
  - 29 maggio ore 15
  - 19 giugno ore 15
  - 11 luglio ore 10
- Iscrizione su AlmaEsami
- **Sufficiente iscrivere un membro del gruppo**
- Se ci sono diverse richieste possiamo fissare un'ulteriore data prima o dopo queste
- **Presentarsi alla discussione con un laptop da usare per discutere il progetto**

# AlgaT

- AlgaT si compone di due parti **obbligatorie**:
  - Simulazione interattiva del funzionamento dell'algoritmo e/o della struttura dati scelta
  - Domande di autoapprendimento
- Ogni gruppo è libero di scegliere:
  - le informazioni che ritiene utili per spiegare l'argomento scelto
  - l'organizzazione complessiva dell'applicazione
  - gli aspetti grafici

# Spiegazione interattiva

- All'apertura AlgaT mostra informazioni di contesto – ad esempio spiegazione testuale dell'algoritmo o *credits* – e permette di far partire il tutorial
- AlgaT non è un tutorial video ma un ambiente interattivo
- Il tutorial include diverse lezioni e l'utente sceglie quale seguire
- **Le lezioni dipendono dall'argomento scelto e sono decise dal gruppo ma devono essere almeno 2**
- In ogni lezione, il tutorial mostra le strutture dati rilevanti e come queste cambiano durante l'esecuzione
- La lezione è organizzata in passi e l'utente controlla l'esecuzione e può scegliere ad esempio quando avanzare, attraverso appositi bottoni dell'interfaccia
  - Granularità, organizzazione e numero di passi sono arbitrari

# Autoapprendimento

- AlgaT prevede anche un'area in cui l'utente può rispondere a domande associate alla lezione
- **Le domande dipendono dall'argomento scelto e sono decise dal gruppo ma devono essere almeno 5 per ogni lezione**
- Ogni gruppo decide il tipo di domande e la difficoltà
- L'applicazione mostra le domande una per volta e verifica la risposta dell'utente, seguita da eventuali spiegazioni
- NON è possibile visualizzare la domanda successiva se l'utente non ha risposto correttamente alla precedente
- Dopo aver risposto alle domande la lezione è conclusa e si torna al menù principale



# Note

- Per semplicità le domande sono pre-caricate nel sistema. Le domande possono essere memorizzate su file
- In generale è importante che sia semplice e chiaro aggiungere nuove domande e nuove lezioni ad AlgaT
- AlgaT deve gestire correttamente tutti gli errori e notificarli all'utente
  - input non corretto, errori di selezione, etc.
- Queste specifiche sono appositamente poco vincolanti, siete liberi di aggiungere funzionalità oltre a quelle descritte finora

# Argomenti

- Ogni gruppo sceglie un argomento tra i seguenti
  1. *Alberi bilanciati di ricerca*
  2. *Tabelle Hash*
  3. *Strutture dati heap e heapsort*
  4. *Cammini minimi*
  5. *Tecniche di progettazione Divide-et-Impera*
  6. *Tecniche di progettazione Greedy*
- Ogni algoritmo può essere scelto da **4 gruppi al massimo** (se non bastano rilasceremo il vincolo)

# Come scegliere e prenotarsi

- Compilare form: <http://diiorio.nws.cs.unibo.it/asd1819/progetto/>
- Non è possibile cambiare gruppo o algoritmo tramite il form, contattare direttamente il docente via mail

Algoritmi e Strutture Dati a.a. 2018/19 - Progetto AlgaT

Info

Gruppo

Nome Gruppo:

Componente 1 - Nome:

Cognome:

Mail Unibo:

Componente 2 - Nome:

Cognome:

Mail Unibo:

Componente 3 - Nome:

Cognome:

Mail Unibo:

Algoritmo

Attenzione:

 il form non richiede conferma prima di spedire i dati. Cliccare su INVIA con la scelta definitiva.

☒ Alberi Bilanciati di Ricerca

☐ Tabelle Hash

☐ Heap e HeapSort

☐ Cammini Minimi

☐ Tecniche di progettazione Divide Et Impera

☐ Tecniche di progettazione Greedy

# Cosa ci aspettiamo dal progetto?

- Il progetto deve:
  - Funzionare
  - Mostrare chiaramente le strutture dati e gli algoritmi usati
  - Gestire gli errori
  - Essere ben strutturato e flessibile
  - Funzionare
- **Soprattutto: deve essere frutto del vostro lavoro, dovete essere in grado di spiegarlo e modificarlo on-the-fly il giorno della consegna!**

# Domande?

- Se ho seguito le lezioni l'anno scorso devo fare il progetto dell'anno scorso o quello di quest'anno?
  - Bisogna consegnare il nuovo progetto, con le nuove regole
  - Chi ha già un voto al progetto dagli anni scorso NON deve consegnare il progetto
- Possiamo usare Java FXML e/o SceneBuilder?  
*Si*
- Possiamo usare Java SWING?  
*Si (ma sconsigliato)*
- Altre domande?