

## Fondamenti di Informatica T2

### **Introduzione al Laboratorio**

*Corso di Laurea in Ingegneria Informatica*

Anno accademico 2021/2022

**Prof. ROBERTA CALEGARI**

**Prof. AMBRA MOLESINI**

*Dipartimento di Informatica – Scienza e Ingegneria (DISI)*



# Struttura del Corso

- **Fondamenti T-2** mira a sviluppare *solide competenze* sia teoriche sia *pratico-operative*
  - non basta "aver studiato": bisogna anche *saper applicare concretamente tale conoscenza, in autonomia, a casi concreti non banali*
  - è molto di più che "scrivere codice": la manualità è indispensabile, ma occorre via via *acquisire esperienza e forma mentis opportuna*
- **Il ruolo del laboratorio è quindi *essenziale***
  - non a caso *l'esame (3 ore!) si svolge interamente in laboratorio*
  - le esercitazioni in lab sono *solo una parte* di ciò che serve: altro laboratorio va svolto a casa, in autonomia: *acquisire una vera competenza richiede moltissimo tempo, non ci sono scorciatoie*
  - NB: 1 credito = 25 ore di lavoro → 12 crediti = 300 ore  
120 in aula e lab, il resto è studio e sviluppo autonomo



# Struttura del Laboratorio

---

- **Per le prime settimane**
  - le esercitazioni saranno rielaborazioni su quanto visto a lezione
  - complessità bassa o medio-bassa
  - vi diremo *cosa fare* e anche (quasi) il dettaglio di *come farlo*
- **Nelle settimane successive**
  - le esercitazioni cominceranno a proporre *casi di studio distinti* dagli esempi visti a lezione
  - complessità media (in aumento..)
  - vi diremo *cosa fare* e una traccia su *come farlo*
- **Da metà corso in poi**
  - proporremo *casi di studio via via più complessi* in domini applicativi variegati, spesso tratti da (parti di) *compiti d'esame*
  - livello di sfida crescente: vi diremo *cosa fare*, ma il come...



# Obiettivi del Laboratorio

- **Sempre: concepire e impostare il software "a oggetti"**
  - Dall'analisi del problema al progetto della soluzione
  - Dal progetto della soluzione al codice che la implementa
  - Il ruolo dell'architettura software: *come e perché* impostarla così?
- **Nelle prime settimane**
  - familiarizzare con il nuovo linguaggio e i relativi strumenti
  - sperimentare i nuovi costrutti visti a lezione (in casi molto simili)
  - assestare le competenze operativo-algoritmiche di Fondamenti T1
- **Nelle settimane successive**
  - affrontare *casi di studio diversi* dagli esempi visti a lezione, capendo come applicare/sfruttare idee e costrutti visti a lezione
  - rafforzare le competenze algoritmiche
  - affrontare via via (parti di) *compiti d'esame*



# No Laboratorio..?

---

- **"Non ho tempo di venire in laboratorio"**
  - "Devo studiare Analisi 2, mi spaventa"
  - "Ho già programmato alle superiori, conosco già anche Java"
  - "Sono stato a lezione e ho capito, quindi non mi serve perdere tempo a far girare gli stessi esempi"
  - "Posso fare le esercitazioni a casa da solo"

# No Laboratorio..?

Distribuzione voti - 16/6/2020



...sicuro..?



Distribuzione voti - 15/06/2021





# Esame in Laboratorio

- **L'esame si terrà in laboratorio (covid permettendo), quindi..**
  - per quanto scomodo possa essere...
  - per quanto sfortunati possano essere i PC...
  - per quanto possiate preferire un Mac o Linux..
- **Occorre familiarizzare *comunque* con i PC dei laboratori e con la loro configurazione**
  - che potrà essere diversa da quella che avete a casa
  - che potrà avere/non avere gli stessi strumenti che avete voi
  - ...l'ingegnere deve comunque cavarsela sempre!
- **..in modo da evitare perdite di tempo durante l'esame**



# Laboratorio: Noi & Voi

---

- Come funziona
  - Noi vi diamo un problema e **VOI** progettate/implementate la soluzione
  - Solo dopo, **NOI** vi facciamo vedere come si sarebbe dovuto risolvere con soluzioni e commenti (e pubblichiamo le soluzioni)
- E se non ci riuscite..?
  - Se siete bloccati chiamate i docenti o i tutor, ma ricordate che lo scopo è *cercare di farcela da soli* (all'esame non potrete chiamare aiuto..)
  - Siete tantissimi: abbiate pazienza, stiamo arrivando 😊
- Regole di comportamento
  - Si rimane fino al termine della lezione (a meno che non finiate – molto – prima..)
  - Si parla a bassa voce (la collaborazione è consentita)
  - Non si mangia/sgranocchia (sono regole dei lab, e di civiltà..)





# Ricordate:

## Non Basta Che Funzioni...

---

- Un buon software dev'essere anche *ben scritto e collaudato*
  - Nel tempo, imparerete a fare l'una e l'altra cosa
- Code style
  - *Scrivere in modo chiaro e leggibile è un MUST*
  - Esistono regole di stile universalmente condivise: rispettare le regole di Java è un preciso requisito ("ah va beh ma tanto.." non è accettabile!)
  - Eclipse vi aiuterà a scrivere codice ben formattato rispettando tali regole
  - *MA non diventate dipendenti dallo strumento!*
- Il docente non è il vostro debugger, né il vostro manuale
  - Se non ricordate una funzione o una libreria, *cercatela, documentatevi!*
  - *Investite sulla vostra autonomia: costa più fatica, ma rende moltissimo*
  - Se una cosa non funziona, noi vi aiutiamo, ma il debugging tocca a voi
  - *Saper usare gli strumenti è parte integrante del vostro compito*



# Problemi...?

## Dubbi Esistenziali..?

---

- Non aspettate che il problema si ingigantisca
  - I problemi e i dubbi vanno affrontati e risolti prima che sia tardi
  - Avete letto le slide? Avete guardato sul sito e su IOL? Avete letto le FAQ?
- Se non capite qualcosa, avete dubbi, vi manca «qualche pezzo» o semplicemente volete fare due chiacchiere:
  - scriveteci (indifferentemente AK/LZ, siamo un team!)

**Prof. Roberta Calegari**

roberta.calegari@unibo.it

Teams\*

*(\*) previo accordo via email*

**Prof. Ambra Molesini**

ambra.molesini@unibo.it

Teams\*

- chiedete un ricevimento "intorno" alle lezioni o in laboratorio
- chiedete appuntamento per un ricevimento ad hoc



# Infine..

---

- Soprattutto, buon lavoro...
- ...e buon divertimento! 😊