

Università degli Studi di Camerino  
Scuola di Scienze e Tecnologie  
Corso di Laurea in Informatica  
Corso di Algoritmi e Strutture Dati 2019/2020  
Parte di Laboratorio (6 CFU)  
Docente: Luca Tesei

## Assegnazione del Miniprogetto 1

### Descrizione

Il progetto consiste nel realizzare l'implementazione di sette classi la cui struttura viene data già impostata:

- TimeSlot rappresenta un intervallo temporale. Per realizzare l'implementazione in maniera corretta occorre studiare (indipendentemente) le [API](#) delle classi `java.util.Calendar` e `java.util.GregorianCalendar` facenti parte della Java Standard Edition;
- Prenotazione: rappresenta una prenotazione di una certa aula in un certo time slot;
- Aula: rappresenta un'aula in cui si può fare lezione o laboratorio. Ha un insieme di facilities e un insieme di prenotazioni;
- Facility: classe astratta che rappresenta una certa facility generica di un'aula
- PresenceFacility: una particolare facility "booleana", cioè indica una facility che può essere presente o no, ad esempio la presenza di un proiettore HDMI;
- QuantitativeFacility: una particolare facility "con quantità", cioè indica una facility associata ad una quantità, ad esempio 40 postazioni con pc o 150 posti a sedere;
- GestoreAule: un attore che mantiene una collezione di aule e permette di fare delle ricerche di aule libere con certe facilities in un certo time slot.

La traccia del codice è fornita come progetto Maven (Apache Maven: <https://maven.apache.org/>) in Eclipse. La versione del compilatore Java definita nel progetto è la 1.8 (Java 8), **non usare versioni superiori**. Sono definiti per il progetto molti metodi di test realizzati con JUnit 5 (<https://junit.org/junit5/docs/current/user-guide/>). I metodi sono definiti per ogni classe da implementare nella cartella `/src/test/java/it/unicam/asdl1920/mp1` all'interno del progetto. L'implementazione dovrebbe passare tutti i test forniti.

### Modalità di Download e Consegna

**Per il download** (si assume l'uso di Eclipse, se si usano altri IDE i passi sono più o meno analoghi):

- scaricare la traccia in formato .zip del progetto maven/eclipse fornito in allegato al post di assegnazione del miniprogetto in Google Classroom;
- scompattare lo .zip in una cartella di propria scelta (la scelta migliore è dentro la cartella del workspace della propria installazione di eclipse);
- da eclipse scegliere File -> Import... poi Selezionare Maven->Existing Maven Projects poi selezionare (tasto Browse...) la cartella scompattata poi dare Finish;
- a questo punto si ha il project in Eclipse e si possono aprire i sorgenti e modificarli e si possono lanciare i test JUnit di ogni classe.

Vanno implementati tutti i metodi richiesti (segnalati con commenti della forma // TODO testo). La specifica precisa delle API è data con commenti javadoc del codice. Non è consentito:

- aggiungere classi pubbliche;
- modificare la firma dei metodi già specificati nella traccia;
- modificare le variabili istanza già specificate nella traccia.

E' consentito:

- aggiungere classi interne private per fini di implementazione;
- aggiungere metodi privati per fini di implementazione;
- aggiungere variabili istanza private per fini di implementazione.

I test forniti possono essere lanciati per controllare che l'implementazione sia corretta.

Nel file sorgente di ogni classe, nel commento javadoc della classe, modificare il campo @author come indicato: "INSERIRE NOME E COGNOME DELLO STUDENTE - INSERIRE ANCHE L'EMAIL [xxxx@studenti.unicam.it](mailto:xxxx@studenti.unicam.it)".

### **Per la consegna:**

Creare una cartella con il seguente nome (lettere maiuscole):

**ASDL1920-NOME-COGNOME-MP1**

ad esempio **ASDL1920-MARIO-ROSSI-MP1**. Nel caso di più nomi/cognomi usare solo il primo nome e il primo cognome. Nel caso di lettere accentate nel nome/cognome usare le corrispondenti lettere non accentate (maiuscole). Nel caso di apostrofi o altri segni nel nome/cognome ometterli. Nel caso di particelle nel nome o nel cognome, ad esempio De Rossi o De' Rossi, attaccarle (DEROSSI).

All'interno di questa cartella copiare i file sorgenti java modificati con la propria implementazione:

- TimeSlot.java
- Prenotazione.java
- Aula.java
- Facility.java
- PresenceFacility.java
- QuantitativeFacility.java
- GestoreAule.java

che si trovano nella cartella

src/main/java/it/unicam/asdl1920/mp1

all'interno della cartella del progetto.

Comprimere la cartella in formato .zip (non rar o altro, solo zip) e chiamare l'archivio

**ASDL1920-NOME-COGNOME-MP1.zip**

**ATTENZIONE:** se i passaggi descritti per la consegna non vengono seguiti

**\*\*precisamente\*\*** (ad esempio nome della cartella sbagliato, contenuto dello zip diverso da quello indicato, formato non zip ecc.) lo studente **perderà automaticamente 3 punti nel voto del miniprogetto.**

Consegnare il file **ASDL1920-NOME-COGNOME-MP1.zip** tramite Google Classroom (usare la funzione consegna associata al post di assegnazione del miniprogetto) entro la data di **scadenza**, cioè **Venerdì 22 Novembre 2019 ore 23.59.**

## Valutazione

La valutazione si baserà sui seguenti criteri, in ordine decrescente di importanza:

**0. Codice scritto individualmente. Nel caso di conclamato “plagio” il miniproject di tutti i “plagi” verrà decurtato di 7 punti.**

1. Numero di test JUnit superati. Nel caso in cui il codice vada in ciclo durante l'esecuzione di un test, il test non riesce a fallire, ma questa situazione viene valutata in maniera peggiore del fallimento del test. Evitare in ogni modo di consegnare se si presenta questa eventualità, piuttosto non implementare il metodo corrispondente (provocando quindi il fallimento del test).
2. Codice chiaro e ben commentato
3. Correttezza di uso delle classi e delle interfacce delle Collections di Java SE.
4. Scelta di strutture dati e implementazione efficienti sia dal punto di vista del tempo di esecuzione che dello spazio richiesto.

Il voto assegnato al miniprogetto verrà comunicato tramite Google Classroom. Il voto sarà espresso in 30esimi e peserà per il 34% del voto finale ottenuto con i miniprogetti. Il miniprogetto 2 peserà per il 33% e il miniprogetto 3 peserà per il restante 33%.