

## Esame di Object Orientation (Gr. 2) – 13 Giugno 2022

- Scrivere immediatamente su ogni foglio che vi è stato consegnato *Cognome, Nome, N° Matricola*.
- Non è consentito consultare appunti, libri, o colleghi, né qualunque dispositivo elettronico, **PENA IMMEDIATO ANNULLAMENTO DELLA PROVA**
- Tempo a disposizione: 3 ore

### Esercizio 1

Si vuole realizzare un sistema informativo per supportare la gestione dell'attività di vendita e noleggio di imbarcazioni usate "Stan's Previously Owned Vessels".

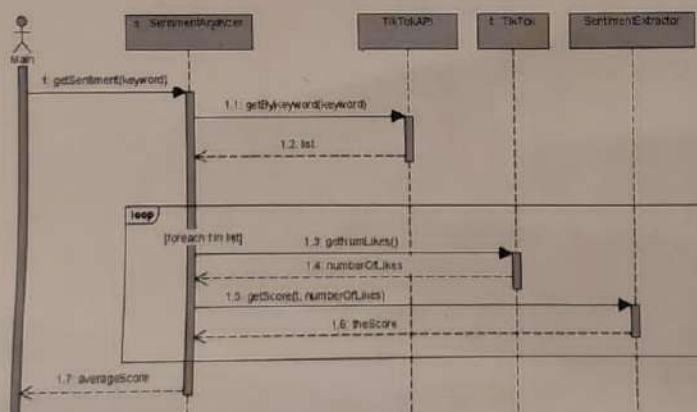
Il sistema permette all'amministratore (Stan) di gestire il parco mezzi aggiungendo, modificando, ed eliminando imbarcazioni. Le imbarcazioni sono identificate da un nome univoco, da una lunghezza (espressa in metri), e possono essere a remi (caratterizzate dal numero di remi), a vela (caratterizzate dal numero di alberi, tipologia di vela ed altro), e a motore (caratterizzate dalla potenza, numero e tipologia dei motori). Inoltre, l'amministratore può anche avviare un'asta per una imbarcazione, indicando una data di termine dell'asta, e una base di partenza (in \$). Gli utenti, previa autenticazione, possono visualizzare le aste in corso e, dopo aver selezionato l'asta di interesse, inviare un'unica offerta indicando la cifra proposta. Allo scadere dell'asta, l'amministratore visualizza l'offerta che si è aggiudicata l'imbarcazione e i dati del compratore.

Si richiede di definire un Class Diagram concettuale con tutte le classi necessarie a modellare la funzionalità di inserimento di un'offerta per un'asta.

Successivamente si richiede di disegnare Class Diagram e Sequence Diagram su una possibile soluzione di questa funzionalità, ad un livello di dettaglio comprendente classi (o Mock-up) dell'interfaccia utente, classi di controllo e classi di modello, eventualmente organizzate in packages basandosi sul pattern Entity-Boundary-Control (EBC).

### Esercizio 2

Si scriva tutto il codice Java che è possibile desumere dal seguente sequence diagram.



### Esercizio 3

```
1 public class Exercise {  
2     public static void main(String args[]) {  
3         String s1 = new String("CIAO");  
4         String s2 = new String("CIAO");  
5         try {  
6             System.out.println("Confronto.");  
7             if (s1 == s2) {  
8                 throw new Exception("Non devono essere uguali!");  
9             }  
10            System.out.println("Confronto effettuato, tutto OK.");  
11        } catch (Exception e) {  
12            System.out.println("Le stringhe erano uguali.");  
13        }  
14        System.out.println("Fatto.");  
15    }  
16 }  
17 }
```

Si consideri il frammento di codice Java mostrato sopra.

1. Il codice compila correttamente? Se no, indicare una possibile correzione del codice affinché compili.
2. Qual è l'output di un'invocazione del metodo main?

### Esercizio 4

Descrivere i concetti di overloading ed overriding, fornendo almeno esempio di ciascuno dei due concetti nel dominio della gestione di un museo. Relativamente agli esempi, si richiede esplicitamente di fornire un class diagram e la relativa implementazione in linguaggio Java, e di indicare in maniera chiara i punti dove sussistono overloading/overriding.