

Ingegneria del Software (6 crediti) a.a. 2011-12

Prova Scritta del 31 agosto 2012

Tempo a disposizione: 2 ore

Svolgere gli esercizi 1+2 e 3+4 su fogli protocollo separati

Esercizio 1

Dato il seguente codice Java relativo alla classe Account (conto corrente) con le usuali operazioni di versamento e prelievo che dipendono dal tipo di conto corrente (nel codice esistono solo due tipi: Famiglia e Azienda).

```
public class Account implements {

    public static final int CONTO_CORRENTE_FAMIGLIA=0;
    public static final int CONTO_CORRENTE_AZIENDA=1;
    private int tipoConto;
    private float saldo;

    public Account(int tipoConto){
        this.saldo=0;
        this.tipoConto=tipoConto;
    }

    public void versamento(float somma) throws Exception{
        if (somma <= 0){
            throw new Exception("Somma rifiutata: " + somma);
        }
        this.saldo = this.saldo + somma;
        // calcola gli interessi di commissione sull'operazione
        float commissioneOperazione = 0;
        switch(tipoConto){
            case Account.CONTO_CORRENTE_FAMIGLIA:
                commissioneOperazione = 10;
                break;
            case Account.CONTO_CORRENTE_AZIENDA:
                commissioneOperazione = 5;
                break;
        }
        // sottraggo la commissione calcolata al saldo
        this.saldo = this.saldo - ((this.saldo * commissioneOperazione)/100);
    }

    public void prelievo(float somma) throws Exception{
        if (somma < 0){
            throw new Exception("Somma negativa: " + somma);
        }
        if (this.saldo < somma){
            throw new Exception("Saldo insufficiente : " + somma);
        }
        this.saldo = this.saldo - somma;
        float commissioneOperazione = 0;
        switch(tipoConto){
            case Account.CONTO_CORRENTE_FAMIGLIA:
                commissioneOperazione = 10;
                break;
            case Account.CONTO_CORRENTE_AZIENDA:
                commissioneOperazione = 5;
                break;
        }
    }
}
```

```

        break;
    }

    this.saldo = this.saldo - ((this.saldo * commissioneOperazione)/100);
}
}

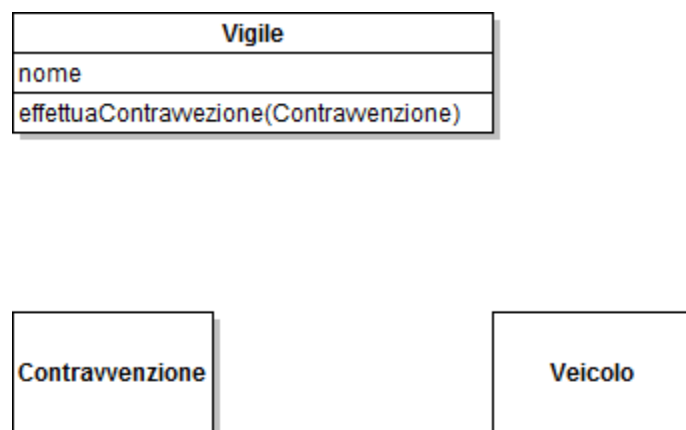
```

- Scrivere dei casi di test JUnit per il metodo **prelievo** (solo per questo metodo) in modo tale da avere 100% statement coverage
- Applicare alla classe Account il refactoring “extract method” per migliorarne la qualità. Chiamare il nuovo metodo **calcolaCommissione()**.
- Sostituire lo switch nel metodo calcolaCommissione() con una soluzione più object oriented. Ovvero applicare il refactoring “replace conditional with polymorphism” dove al posto dello switch introduciamo il polimorfismo basato sulla classe Account.

Esercizio 2

L'applicazione da progettare riguarda le informazioni sulle contravvenzioni stradali irrogate dalla Polizia Municipale per violazione del Codice della Strada. Un **vigile**, che ha un nome, un cognome e una matricola, può effettuare zero o più contravvenzioni. Una **contravvenzione** ha un numero (numero_verbale) che è un intero ed una descrizione nella quale viene inserito il luogo dove è avvenuta la contravvenzione. Inoltre una contravvenzione è relativa ad un veicolo il quale ha una targa, un colore e un tipo_veicolo (es. auto o moto). Può anche accadere che allo stesso veicolo vengano irrogate più contravvenzioni. Infine ad ogni veicolo corrisponde un proprietario che può essere una persona fisica o una persona giuridica (ovvero una società).

- Completare il seguente diagramma UML seguendo la descrizione data sopra.



- Scegliere il diagramma UML più opportuno per descrivere la seguente procedura di pagamento contravvenzioni da parte di un'utente. L'utente ricevuta la contravvenzione ha tempo 30 giorni per pagare la multa o fare ricorso. L'utente di fatto può scegliere se pagare on-line (caso 1), pagare direttamente di persona in municipio (caso 2) oppure fare ricorso (caso 3). Nel primo caso l'utente deve effettuare la registrazione al sito www.xyz.it (se non già iscritto) inserendo il proprio indirizzo e-mail. Gli utenti registrati potranno effettuare il pagamento, accedendo alla pagina Web “Paga contravvenzioni”. Dopo di che, l'utente dovrà inserire tutti i dati richiesti: importo da pagare; cognome e nome; indirizzo; CAP; città; provincia; numero verbale; targa; data verbale. e scegliere la modalità di pagamento, ovvero: carta di credito (VISA e MasterCard); carta PostePay; BancoPostaonline. A questo punto, cioè prima di avere effettuato realmente il pagamento, il sistema fornirà una schermata riassuntiva e l'utente dovrà controllare l'esattezza dei dati inseriti. In caso di

errore l'utente dovrà ri-inserire tutti i dati a partire dall'importo. Altrimenti l'utente può confermare e il pagamento viene effettuato. Nel caso 2 (pagamento in municipio) l'utente compilerà in municipio un modulo dove verranno inseriti tutti i dati necessari (gli stessi che devono essere inseriti via Web) e potrà scegliere se pagare "cash" direttamente al vigile o tramite bollettino postale. Infine nel caso di ricorso (caso 3) ci sono due opzioni: ricorso al **Prefetto** oppure ricorso al **Giudice di Pace**. Il ricorso può essere presentato mediante raccomandata con ricevuta di ritorno o mediante consegna a mano presso l'ufficio competente. Al ricorso dovrà essere allegato il verbale della contravvenzione. Il ricorso e la documentazione allegata dovranno essere presentati in originale unitamente ad ulteriori cinque copie.

Esercizio 3

Il sistema Telepass come noto è un servizio che consente, attraverso l'utilizzo di un piccolo trasmettitore collocato sul parabrezza dell'auto o posizionato sulla moto, di transitare nelle porte dedicate Telepass di Autostrade per l'Italia e di pagare il pedaggio per la tratta percorsa senza fermarsi alla barriera. Il trasmettitore Telepass, assegnato a un contraente che definisce la forma di pagamento, può essere utilizzato su un massimo di tre veicoli, intestati al contraente o a suoi familiari, le cui targhe devono essere comunicate ad Autostrade per l'Italia. I pedaggi vengono addebitati periodicamente su un conto, che può essere un conto corrente, un BancoPosta o una carta di credito abilitata.

Il funzionamento pratico è relativamente semplice: quando un veicolo si avvicina a una porta, un impianto ottico (CTV) riconosce il tipo del veicolo, classificandolo in una delle cinque classi tariffarie e attiva l'emissione, da parte della porta, del segnale di richiesta di identificazione. Il trasmettitore presente sull'auto risponde al segnale del trasmettitore a terra, ritrasmettendo un codice identificativo univoco. Trascorso mezzo secondo, la centralina a terra dà ordine di alzare la sbarra. Se, nel frattempo, ha ricevuto dal trasmettitore un codice corretto, registra il passaggio regolare.

Altrimenti, una speciale fotocamera fotografa la targa, per poter risalire all'autore del transito irregolare.

- a) Identificare gli attori del sistema, distinguendo in attori primari e secondari.
- b) Identificare i casi d'uso del sistema e produrre un diagramma dei casi d'uso.
- c) Specificare il caso d'uso corrispondente all'operazione di transito del veicolo.

Esercizio 4

Si consideri un semplicissimo editor di testo in cui i documenti (classe **Document**) sono rappresentati come **ArrayList** di linee di testo (stringhe), ed è possibile l'aggiunta e la cancellazione di linee tramite i metodi **write(String text)** e **erase(int lineNbr)** [che semplicemente invocano i metodi **add** e **remove** di **ArrayList** con gli stessi parametri] nonché la visualizzazione del contenuto del documento. Si vuole estendere tale editor con la possibilità di effettuare undo e redo di operazioni di aggiunta e cancellazione di linee di testo.

- a) Identificare un Design Pattern che potrebbe essere istanziato per realizzare il sistema precedente.
- b) Produrre un diagramma delle classi che istanzi il Design Pattern individuato sul sistema in esame, mettendo in corrispondenza le classi dell'applicazione con quelle del Design Pattern originario.
- c) Discutere i principali vantaggi apportati dall'uso del Design Pattern nella realizzazione del sistema.