

Ingegneria del Software a.a. 2014-15

Prova Scritta del 13 gennaio 2015

Esercizio di sbarramento

COGNOME

NOME

MATRICOLA

Rispondere alle seguenti domande. Per ogni domanda, solo una soluzione è corretta. L'esercizio si ritiene superato se si risponde correttamente ad **almeno 6 domande**, la valutazione è di 1 punto per ogni risposta corretta oltre le 6.

Domanda 1

In qualità di project manager, quale tra i seguenti modelli di processo di sviluppo software utilizzeresti per progettare un sistema safety-critical medicale di radioterapia molto complesso ed affidabile avendo a disposizione 40 sviluppatori?

- a) UML
- b) Extreme programming
- c) Test Driven Development (TDD)
- d) Unified Process

Domanda 2

Solitamente la **convalida** (o validazione) dei requisiti si effettua mediante *formal peer reviews*. Un altro modo per effettuare la convalida dei requisiti è:

- a) Utilizzare la tecnica del brainstorming
- b) Chiedere a degli analisti (che non sono stati coinvolti nella stesura dei requisiti) di effettuare un'analisi dettagliata dei requisiti con lo scopo di identificare possibili aree di miglioramento
- c) Scrivere dei casi di test a partire dai requisiti
- d) Cercare eventuali requisiti contraddittori (o in conflitto) ed identificare i missing requirements

Domanda 3

Qual è il significato di "analisi dei trade-off" nel contesto delle architetture software?

- a) Valutare i vantaggi e svantaggi di una particolare architettura software tenendo in considerazione gli attributi o proprietà di qualità di sistema (es. Performance)
- b) Verificare che l'architettura software scelta soddisfi i requisiti software
- c) Analizzare l'architettura software con il fine di valutarne la tracciabilità con il codice
- d) Valutare che l'architettura software sia conforme ai principi di buona progettazione/programmazione (cioè: astrazione, modularità, alta coesione, information hiding, basso accoppiamento tra moduli, semplicità)

Domanda 4

Quanti casi di test sono necessari per avere 100% **Multiple Condition Coverage (MCC)** del seguente pseudo-codice?

```
1 READ A
2 READ B
3 C = A - 2 * B
4 IF C < 0 THEN
5   PRINT "C negative"
6 ENDIF
```

- a) uno
- b) tre
- c) due
- d) zero

Domanda 5

Quale tra i diagrammi messi a disposizione da UML è il più indicato per rappresentare/descrivere/modellare un'operazione di una classe?

- a) Activity diagram
 - b) Sequence diagram
 - c) State machine diagram
 - d) Operation specification diagram
-

Domanda 6

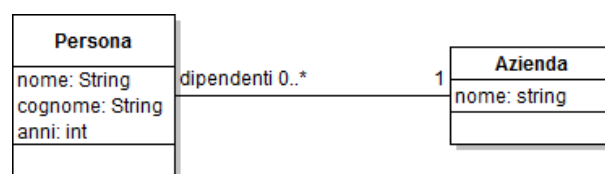
In termini di principi OO di progettazione che cosa si può affermare della seguente classe **Car**?

```
public class Car {  
    private String make = "Aston Martin";  
    private int numberOfWindows = 5;  
  
    public void drive() {  
        // do some driving!  
    }  
  
    public void brake() {  
        // slow down!  
    }  
  
    public void stop() {  
        // STOP!!!!  
    }  
  
    public void printDocument(Document d) {  
        // print Document!!!!  
    }  
}
```

- a) La classe Car è fortemente accoppiata
 - b) L'information hiding della classe Car è basso
 - c) La classe Car ha coesione alta
 - d) La classe Car ha coesione bassa
-

Domanda 7

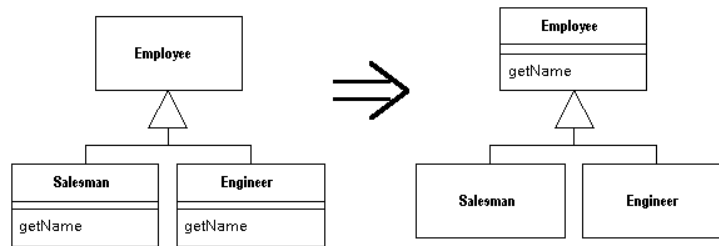
Dato il seguente class diagram quale delle seguenti espressioni OCL esprime il vincolo: “ogni azienda non deve avere più di mille dipendenti”?



- a) Context Azienda inv: self.dipendenti->size() <= 1000
 - b) Context Persona inv: dipendenti <= 1000
 - c) Context Azienda inv: self.nome.dipendenti->size() <= 1000
 - d) Context Azienda::dipendenti inv: self.dipendenti->size() < 1000
-

Domanda 8

Come si chiama il refactoring applicato nell'esempio seguente?



- a) Move fields
- b) Pull-up method
- c) Extract class
- d) Extract method

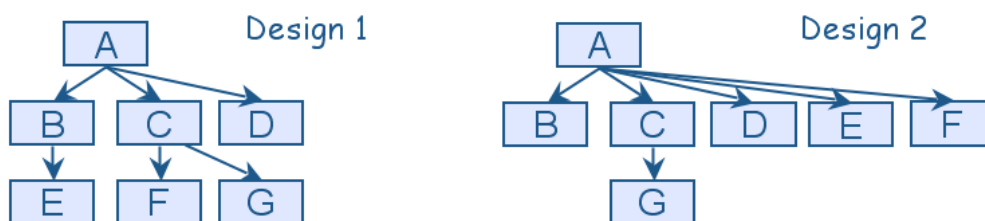
Domanda 9

Utilizzando il metodo di sviluppo Extreme Programming come si produce di solito il design iniziale di un applicazione?

- a) Si utilizza un UML modeller (ad esempio Visual Paradigm) e si progetta un blueprint (progetto dettagliato) utilizzando la textual analysis a partire dalle user stories
- b) Si utilizza il metodo CRC (Class Responsibility Collaboration), che consiste nell'utilizzare dei foglietti adesivi sui quali si scrive il nome della classe, lo scopo e le collaborazioni con le altre classi
- c) Il design iniziale non viene mai prodotto in quanto uno dei principi di Extreme programming è proprio quello di "saltare" la fase di design e "passare" direttamente dai requisiti alla fase di codifica
- d) Il design viene prodotto partendo dal modello del dominio ed arricchendolo fino ad ottenere un "modello eseguibile". Successivamente il modello viene trasformato in codice usando un'apposito tool

Domanda 10

Si supponga di avere due design differenti che realizzano lo stesso sistema software. Quale dei due è preferibile e perchè?



- a) Il Design 2 è preferibile al Design 1 perchè esiste un solo modulo (G) che ha distanza 2 dalla radice. Invece nel Design 1 ci sono tre moduli (E, F e G) a distanza 2 dalla radice
- b) Il Design 1 è preferibile al Design 2 perchè l'accoppiamento tra moduli totale è minore
- c) Il Design 1 è preferibile al Design 2 perchè il modulo A del Design 2 ha fan-out=5 (alto) che indica un'eccessiva dipendenza
- d) Il Design 2 è preferibile al Design 1 perchè è un istanza dello stile architetturale Call-Return model