

## Progetto del software

### Simulazione prova scritta – 29 maggio 2024 – 2h

#### PARTE 1 – RISPOSTA SINGOLA - Ogni domanda ha una sola risposta VERA.

- Una risposta esatta fa acquisire il punteggio positivo riportato a fianco della domanda
- Una risposta errata fa perdere il punteggio negativo riportato a fianco della domanda
- Una risposta lasciata in bianco viene valutata 0

1. **(3, -.5)** Il design pattern *singleton*
  - a) Non può essere usato nella programmazione ad oggetti
  - b) **Può essere usato nella programmazione ad oggetti**
  - c) E' sempre utile nella programmazione ad oggetti
  - d) E' utile quando si debbono creare esattamente tre istanze di uno stesso oggetto
2. **(3, -.5)** L'utilizzo di UML è fondamentale perché
  - a) **Consente di modellare un software al livello di astrazione necessario alla fase del progetto in cui si sta lavorando**
  - b) Obbliga a modellare un software in termini di numero di persone che ci possono lavorare
  - c) Non consente mai ambiguità
  - d) Impedisce l'utilizzo della programmazione funzionale, che è complessa da usare
3. **(3, -.5)** Il modello di sviluppo denominato *Waterfall*
  - a) **Prevede una fase di specifica dei requisiti, ed una di test/assessment di tali requisiti**
  - b) Prevede una fase di specifica dei requisiti, ma non una di test/assessment di tali requisiti
  - c) Non prevede la fase di specifica dei requisiti
  - d) Non prevede una fase di specifica dei requisiti, né una di test/assessment di tali requisiti

## PARTE 2 – (POSSIBILI) RISPOSTE MULTIPLE -

Ogni domanda può avere da zero a quattro risposte CORRETTE.

- Ogni risposta esatta viene calcolata: +1
  - Ogni risposta errata viene calcolata: -0.5
  - Una risposta lasciata in bianco viene calcolata: 0
4. I diagrammi UML di sequenza
- a) Modellano classi ed oggetti
  - b) Modellano una linea temporale di uno o più specifici caso d'uso
  - c) Non possono venire utilizzati nello sviluppo del SW, ma solo per la documentazione
  - d) Modellano sia chiamate sincrone che asincrone
5. Un tool come Git
- a) Consente a più persone di lavorare contemporaneamente sulla stessa riga di codice senza che si verifichino problemi
  - b) E' utile in ambienti "chiusi", dove ognuno lavora da solo, e non collabora con altri sviluppatori
  - c) Non consente il deploy degli artefatti software costruiti
  - d) Impone di fare un "git add" prima di un "git commit", in maniera esplicita o implicita
6. Le principali difficoltà, quando si interagisce con un *Customer*
- a) Sono dovute al fatto che spesso non sa neppure lui cosa vuole
  - b) Sono dovute al fatto che abita lontano da noi
  - c) Sono dovute al fatto che potrebbe non avere il nostro stesso background, pertanto si verificheranno sicuramente incomprensioni
  - d) Nessuna delle precedenti
7. Le proprietà non funzionali del software
- a) Sono esplicitate sempre dal *customer*
  - b) Sono raramente esplicitate dal *customer*
  - c) Sono, ad esempio, la manutenibilità, scalabilità, interoperabilità
  - d) Non impattano sul processo di sviluppo
8. Un approccio modulare *top-down*
- a) Esclude a priori l'utilizzo di componenti e librerie pre-esistenti
  - b) Parte da una visione di basso livello, dei e librerie pre-esistenti
  - c) Parte da una visione di alto livello
  - d) Talvolta può richiedere di partire da zero

### **PARTE 3 – DOMANDE APERTE**

- Una risposta esatta fa acquisire il punteggio positivo riportato a fianco della domanda
  - Una risposta errata può eventualmente causare una penalità che dipende dalla gravità dell'errore
  - Una risposta lasciata in bianco viene calcolata: 0
  - L'eventuale sfioramento del limite di righe o parole (laddove imposto), porterà a una decurtazione di un punto per ogni riga. Eventuali schematici e listati di codice non verranno presi in considerazione nel calcolo delle righe
  - **SI RICORDA CHE L'UNICO FOGLIO DA CONSEGNARE E' IN CALCE AL COMPITO. QUESTO FOGLIO, PUO' SERVIRE ESCLUSIVAMENTE COME "BRUTTA COPIA". EVENTUALI RISPOSTE SCRITTE IN QUESTO FOGLIO NON VERRANNO PRESE IN CONSIDERAZIONE**
9. **(8 pt)** Si descriva esaurientemente il modello architetturale asincrono, ad esempio, per un'architettura server distribuita

Nome e Cognome \_\_\_\_\_ Matricola: \_\_\_\_\_

**10. (4 pt)** Si descriva brevemente (massimo 5 righe) a cosa servono i design pattern

Nome e Cognome \_\_\_\_\_ Matricola: \_\_\_\_\_

## Progetto del software

### Simulazione prova scritta – 29 maggio 2024 – 2h

Indicare le risposte corrette apponendo una croce nella casella corrispondente. Per superare la prova bisogna aver raggiunto almeno 9 punti nelle domande a risposta singola/multipla, ed almeno 15 complessivamente. Questa è l'unica pagina che dovete consegnare.

	Risposte				Punti/ Penalità	
	A	B	C	D		
1					3	-0.5
2					3	-0.5
3					3	-0.5
4						
5						
6						
7						
8						

Risposta alla domanda 9 (8 pt):

Nome e Cognome \_\_\_\_\_ Matricola: \_\_\_\_\_

**Risposta alla domanda 10 (4 pt):**