Nome e Cognome _____ Matricola: _____

Progetto del software
Prova scritta – 9 giugno 2025 – 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)** In un sistema ad oggetti, la generalizzazione/specializzazione:
   a) Modella la relazione “padre-figlio” di due o più classi
   b) Modella la relazione “produttore-consumatore” di due o più classi
   c) Modella la relazione “genero-suocero” di due o più classi
   d) La domanda è sbagliata, in quanto la “generalizzazione/specializzazione” non è un concetto applicabile ai sistemi ad oggetti

2. **(3, -.5)** L’analisi dei requisiti insieme al “cliente”:
   a) Viene tipicamente fatta in una solo incontro, a valle del quale i requisiti possono cambiare
   b) Non consente di modellare requisiti di performance
   c) Non consente di modellare requisiti di reliability
   d) Nessuna delle precedenti

3. **(3, -.5)** Il principio *Interface segregation* della programmazione SOLID:
   a) Si applica solo alla programmazione funzionale
   b) Prevede che le interfacce progettate siano il più possibile minimali
   c) Aiuta a progettare l’ereditarietà fra le classi di un sistema software
   d) Rende il codice più robusto, ma meno scalabile

Nome e Cognome _____ Matricola: _____

PARTE 2 – (POSSIBILI) RISPOSTE MULTIPLE -
Ogni domanda può avere <u>da zero a quattro</u> risposte CORRETTE.

- **Ogni risposta esatta viene calcolata: +1**
- **Ogni risposta errata viene calcolata: -0.5**
- **Una risposta lasciata in bianco viene calcolata: 0**

4. Una chiamata asincrona:
   a) Generalmente prevede una qualche forma di *callback* per gestire il corretto/l'errato funzionamento della chiamata
   b) Non prevede funzioni di *callback*, perché la chiamata stessa ritorna un codice che rappresenta la corretta (o meno) avvenuta esecuzione dell'operazione
   c) Generalmente ritorna un codice che rappresenta la corretta (o meno) presa in carico dell'operazione
   d) Può avvenire in modalità *fire-and-forget*, ed in quel caso non si hanno garanzie dirette della sua corretta presa in carico da parte del server

5. Il Technology Readiness Level (TRL):
   a) Non è definibile al di fuori della programmazione ad oggetti
   b) E' definibile solo nei sistemi che hanno interfacce scritte in Java
   c) Esiste per definire l'aderenza allo standard IEEE 803 (SRS)
   d) E' un numero da 1 a 9

6. Il design pattern Hardware Proxy / Hardware Abstraction Layer:
   a) E' utile nei sistemi embedded, anche se potrebbe causare un minimo deterioramento delle prestazioni
   b) Non si usa nei sistemi embedded, perché ne risentirebbero le prestazioni
   c) Aiuta a scrivere software in maniera più veloce
   d) Consente di "nascondere" il dispositivo specifico in uso attraverso un'interfaccia software di più alto livello, detta proxy

7. L'utilizzo di tool per il versioning come Git:
   a) E' l'unico modo per riuscire a gestire progetti di grandi dimensioni
   b) Non si applica a team di piccole dimensioni
   c) Impatta sul processo di sviluppo
   d) Può rappresentare uno svantaggio per il team, a lungo termine, ed infatti si usa solo nelle fasi prototipali di un software

8. La *dependency injection*:
   a) Consente di testare un modulo software fornendo dei *mock* delle sue dipendenze
   b) Consente di testare un modulo software solo se è di piccole dimensioni
   c) E' incompatibile con l'uso di librerie pre-esistenti
   d) Consente di massimizzare la *code coverage* del *system under test* (SUT) in quanto il codice dello stesso non cambia

Nome e Cognome _____ Matricola: _____

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 sforamento 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 describa le principali caratteristiche di un linguaggio *managed*, nel caso portando anche esempi di linguaggi visti nel corso

Nome e Cognome _____ Matricola: _____

10. **(3 pt)** Si definisca un un concetto a piacere fra quelli visti nel corso (es: analisi dei requisiti, definizione di design pattern, solo per fare alcuni esempi). In massimo 5 righe.

Nome e Cognome _____ Matricola: _____

Progetto del software
Prova scritta – 9 giugno 2025 – 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 (3 pt):