

Simulazione d'Esame - Risposte

Simulazione d'Esame set D

Versione 1.3

ISTQB® Certified Tester Syllabus Foundation Level

Compatibile con il Syllabus versione 4.0

International Software Testing Qualifications Board



Avviso di copyright

Avviso sul Copyright © International Software Testing Qualifications Board (di seguito chiamato ISTQB®).

ISTQB® è un marchio registrato di International Software Testing Qualifications Board. Tutti i diritti riservati.

Gli autori dichiarano con la presente di trasferire il copyright a International Software Testing Qualifications Board (ISTQB®). Gli autori (come attuali titolari del copyright) e ISTQB® (come futuro titolare del copyright) hanno concordato le seguenti condizioni di utilizzo:

Possono essere copiati estratti di questo documento per un uso non commerciale se la fonte è riconosciuta.

Qualsiasi azienda Training Provider accreditata alla Formazione può utilizzare questa Simulazione d'Esame nei loro corsi di Formazione se gli autori e ISTQB® sono riconosciuti come fonti e proprietari del copyright della Simulazione d'Esame, e a condizione che qualsiasi pubblicità di tale corso di formazione sia fatta dopo che l'accREDITAMENTO ufficiale dei materiali di formazione è stato ricevuto da un Member Board riconosciuto da ISTQB®.

Qualsiasi individuo o gruppo di individui può utilizzare questa Simulazione d'Esame, se gli autori e ISTQB® sono riconosciuti come fonti e possessori del copyright di questa Simulazione d'Esame.

È proibito qualsiasi altro utilizzo di questa Simulazione d'Esame senza prima avere ottenuto l'approvazione scritta di ISTQB®.

Qualsiasi Member Board riconosciuto da ISTQB® può tradurre questa Simulazione d'Esame a condizione di riprodurre il sopra menzionato Avviso di Copyright nella versione tradotta della Simulazione d'Esame.

Responsabilità del documento

Exam Working Group ISTQB® è responsabile di questo documento.

Questo documento è mantenuto da un team dedicato ISTQB® che consiste nel Syllabus Working Group e nell'Exam Working Group.

Riconoscimenti

Questo documento è stato prodotto da un core team di ISTQB®: Stuart Reid e Adam Roman

Il core team ringrazia Exam Working Group, Syllabus Working Group e i Member Board per i loro suggerimenti e contributi.

Storia delle revisioni

Simulazione d'esame – Modello Layout Risposte utilizzato: Versione 2.11 Data: 16 ottobre 2023

Versione	Data	Osservazioni
1.3	20 marzo 2024	Versione allineata alla versione inglese V1.3 (20 marzo 2024). Correzione della domanda: #8, #16
1.0	16 ottobre 2023	Prima versione

Indice dei contenuti

Avviso di copyright	2
Responsabilità del documento.....	2
Riconoscimenti	2
Storia delle revisioni	3
Indice dei contenuti	4
Introduzione	5
Scopo del documento	5
Istruzioni	5
Chiave di risposta.....	6
Risposte	7
1	7
2	8
3	9
4	10
5	11
6	12
7	13
8	14
9	15
10	16
11	16
12	17
13	18
14	19
15	20
16	21
17	22
18	23
19	23
20	24
21	24
22	25
23	26
24	27
25	28
26	28
27	29
28	29
29	30
30	30
31	31
32	31
33	32
34	32
35	32
36	33
37	34
38	35
39	36
40	37

Introduzione

Scopo del documento

Le domande di questa simulazione, le risposte e le relative giustificazioni associate alle domande di questa simulazione d'esame sono state create da un team di esperti in materia e da scrittori esperti di domande, con l'obiettivo di:

- Supportare i Member Board e l'Exam Board ISTQB® nelle loro attività di scrittura delle domande.
- Fornire ai Training Provider e ai candidati all'esame esempi di domande d'esame.

Queste domande non possono essere utilizzate senza essere modificate in nessun esame ufficiale.

Si noti che gli esami reali possono includere un'ampia varietà di domande, e questa simulazione d'esame **non** intende includere esempi di tutti i possibili tipi, stili o lunghezze di domande; inoltre, questa simulazione d'esame potrebbe essere più difficile oppure meno difficile di qualsiasi esame ufficiale.

Istruzioni

In questo documento potete trovare:

- Tabella delle Risposte Chiave, includendo per ogni risposta corretta:
 - Livello di conoscenza (Livello K), obiettivo di apprendimento (LO) e valore del punteggio
- Insieme di risposte, includendo per tutte le domande:
 - Risposta corretta
 - Giustificazione per ogni opzione di risposta
 - Livello di conoscenza (Livello K), obiettivo di apprendimento (LO) e valore del punteggio
- Insieme di risposte aggiuntive, includendo per tutte le domande [non si applica a tutte le simulazioni d'esame]:
 - Risposta corretta
 - Giustificazione per ogni opzione di risposta
 - Livello di conoscenza (Livello K), obiettivo di apprendimento (LO) e valore del punteggio
- *Le domande sono contenute in un documento separato*

Chiave di risposta

Numero domanda (#)	Risposta corretta	LO	Livello K	Punti
1	d	FL-1.1.1	K1	1
2	c	FL-1.2.3	K2	1
3	a	FL-1.3.1	K2	1
4	b	FL-1.4.1	K2	1
5	a	FL-1.4.3	K2	1
6	d	FL-1.4.5	K2	1
7	a	FL-1.5.2	K1	1
8	b	FL-1.5.3	K2	1
9	a	FL-2.1.2	K1	1
10	a	FL-2.1.3	K1	1
11	d	FL-2.1.4	K2	1
12	b	FL-2.1.6	K2	1
13	a	FL-2.2.2	K2	1
14	b	FL-2.3.1	K2	1
15	c	FL-3.1.1	K1	1
16	c	FL-3.1.2	K2	1
17	b	FL-3.2.2	K2	1
18	b	FL-3.2.3	K1	1
19	b	FL-4.1.1	K2	1
20	b, e	FL-4.2.1	K3	1

Numero domanda (#)	Risposta corretta	LO	Livello K	Punti
21	c	FL-4.2.2	K3	1
22	a	FL-4.2.3	K3	1
23	b	FL-4.2.4	K3	1
24	c	FL-4.3.1	K2	1
25	a	FL-4.3.3	K2	1
26	c	FL-4.4.1	K2	1
27	d	FL-4.4.2	K2	1
28	d	FL-4.5.1	K2	1
29	a	FL-4.5.3	K3	1
30	b, d	FL-5.1.3	K2	1
31	a	FL-5.1.4	K3	1
32	b	FL-5.1.5	K3	1
33	c	FL-5.1.7	K2	1
34	b	FL-5.2.1	K1	1
35	b, e	FL-5.2.2	K2	1
36	c	FL-5.3.2	K2	1
37	d	FL-5.4.1	K2	1
38	a	FL-5.5.1	K3	1
39	b	FL-6.1.1	K2	1
40	c	FL-6.2.1	K1	1

Risposte

Numero domanda (#)	Risposta corretta	Spiegazione / Motivazione	Obiettivo di apprendimento (LO)	Livello K	Numero di punti
1	d	a) Non è corretta. Rilevare e correggere difetti nell'oggetto di test non è un obiettivo tipico del test. Sebbene rilevare difetti sia un obiettivo del testing, correggere difetti non è un'attività di test b) Non è corretta. Mantenere una comunicazione efficace con gli sviluppatori non è un obiettivo tipico del test. Sebbene sia utile per raggiungere altri obiettivi del test, ad esempio fornire agli stakeholder informazioni che consentano di prendere decisioni informate, non è un motivo principale per eseguire il testing c) Non è corretta. Validare che i requisiti legali siano stati soddisfatti non è un tipico obiettivo del test, perché la validazione si occupa di verificare se il sistema soddisfa le esigenze degli utenti e di altri stakeholder nel suo ambiente operativo. Verificare che i requisiti legali siano stati soddisfatti è una forma di verifica d) È corretta. Creare fiducia nella qualità dell'oggetto di test viene ottenuta dall'esecuzione dei test che passano	FL-1.1.1	K1	1

Numero domanda (#)	Risposta corretta	Spiegazione / Motivazione	Obiettivo di apprendimento (LO)	Livello K	Numero di punti
2	c	<p>a) Non è corretta. Il calcolo errato dei bonus è un failure di sistema, non un difetto</p> <p>b) Non è corretta. Il sistema che non supporta adeguatamente gli utenti disabili è un failure che alla fine si traduce in una multa, ma la multa stessa non è un failure (sembra essere il corretto funzionamento del sistema normativo)</p> <p>c) È corretta. L'errore è stato commesso dal programmatore e questo errore è stato causato dal fatto che ha lavorato sotto una forte pressione sui tempi, che è la root cause del successivo difetto</p> <p>d) Non è corretta. La cattiva progettazione dell'interfaccia utente, che non indirizza adeguatamente gli utenti disabili, è un difetto di progettazione causato da un errore del progettista. Pertanto, la progettazione dell'interfaccia utente include un difetto di progettazione, non un errore del progettista</p>	FL-1.2.3	K2	1

Numero domanda (#)	Risposta corretta	Spiegazione / Motivazione	Obiettivo di apprendimento (LO)	Livello K	Numero di punti
3	a	<p>a) È corretta. Il principio "i test perdono di efficacia" riguarda l'idea che ripetere test identici su codice non modificato è improbabile che possa rilevare nuovi difetti, e quindi modificare i test può essere fondamentale. Se si utilizzano condizioni di test di alto livello per generare ogni volta nuovi test, i test non saranno identici e non dovrebbero "perdere di efficacia"</p> <p>b) Non è corretta. Il principio "l'assenza di difetti è un'idea sbagliata" riguarda la necessità di garantire che le esigenze degli utenti siano soddisfatte anche se è stato eseguito molto testing e non si rilevano difetti (cioè, è necessaria anche la validazione). L'uso di condizioni di test di alto livello per generare test case ed eseguire test non indirizza direttamente questo problema</p> <p>c) Non è corretta. Il principio "il testing anticipato permette di risparmiare tempo e denaro" riguarda la correzione anticipata dei difetti per prevenire la comparsa di difetti successivi nei prodotti di lavoro derivati, riducendo così i costi e la probabilità di failure. Questo viene tipicamente indirizzato iniziando il testing (sia statico che dinamico) il prima possibile, ma non viene indirizzato utilizzando condizioni di test di alto livello per generare test case ed eseguire i test</p> <p>d) Non è corretta. Il principio "i difetti si raggruppano in cluster" riguarda la distribuzione dei difetti in un sistema, che in genere segue una distribuzione di Pareto. L'uso di condizioni di test di alto livello per generare test case ed eseguire test non indirizza questo problema, che è generalmente indirizzato dal testing basato sul rischio</p>	FL-1.3.1	K2	1

Numero domanda (#)	Risposta corretta	Spiegazione / Motivazione	Obiettivo di apprendimento (LO)	Livello K	Numero di punti
4	b	<p>Si considerino le attività di test elencate e i relativi compiti:</p> <p>A. Analisi dei test - Per identificare le funzionalità che richiedono il testing, viene analizzata la base di test e vengono definite le condizioni di test, che sono poi priorizzate insieme ai rischi correlati. Durante l'analisi dei test, vengono di solito rilevati i difetti nella base di test e può anche essere valutata la testabilità dell'oggetto di test. (Compito 4)</p> <p>B. Progettazione dei test - Vengono utilizzate le condizioni di test per creare i test case e altro testware necessario, come i requisiti dei dati di test e i test charter per il testing esplorativo. (Compito 1)</p> <p>C. Implementazione dei test - Le test procedure, come i test script manuali e automatizzati, vengono create dai test case e possono essere raggruppate in test suite. Le test procedure vengono priorizzate e organizzate in una schedulazione di esecuzione dei test. (Compito 3)</p> <p>D. Completamento dei test - Viene eseguita in specifiche milestone di progetto, come il rilascio, la fine dell'iterazione o la fine del livello di test. Il testware viene identificato e archiviato oppure consegnato ai team appropriati per il riutilizzo, l'ambiente di test viene consolidato e le attività di test vengono analizzate attraverso lessons learned e miglioramenti futuri. (Compito 2)</p> <p>Quindi:</p> <p>a) Non è corretta b) È corretta. La corrispondenza CORRETTA è: 1B, 2D, 3C, 4A c) Non è corretta d) Non è corretta</p>	FL-1.4.1	K2	1

5	a	<p>Si consideri ogni elemento del testware elencato e l'attività di test che lo genera:</p> <ul style="list-style-type: none"> i. Il test completion report è un output dell'attività di completamento dei test ii. I dati memorizzati in un database che viene utilizzato per gli input dei test e per i risultati attesi sono i dati di test – un output dell'attività di implementazione dei test iii. La lista degli elementi necessari per costruire l'ambiente di test è la lista dei requisiti dell'ambiente di test – un output dell'attività di progettazione dei test iv. Le sequenze documentate dei test case in ordine di esecuzione sono test procedure – un output dell'attività di implementazione dei test v. Test case – un output dell'attività di progettazione dei test <p>L'implementazione dei test produce i seguenti output: test procedure (iv), test script automatizzati, test suite, dati di test (ii), schedulazione di esecuzione dei test ed elementi dell'ambiente di test quali stub, driver, simulatori e virtualizzazioni dei servizi.</p> <p>Quindi:</p> <ul style="list-style-type: none"> a) È corretta. Gli elementi ii e iv nella lista sono prodotti come risultato dell'implementazione dei test b) Non è corretta c) Non è corretta d) Non è corretta 	FL-1.4.3	K2	1
---	---	--	----------	----	---

6	d	<p>a) Non è corretta. Il ruolo del testing è principalmente responsabile degli aspetti tecnici del testing, come l'analisi dei test, la progettazione dei test, l'implementazione dei test e l'esecuzione dei test. Valutare la base di test per rilevare difetti e valutare l'oggetto di test per la testabilità sono compiti eseguiti come parte dell'analisi dei test, quindi è probabile che siano compiti svolti dal ruolo del testing</p> <p>b) Non è corretta. Il ruolo del testing è principalmente responsabile degli aspetti tecnici del testing, come l'analisi dei test, la progettazione dei test, l'implementazione dei test e l'esecuzione dei test. Definire i requisiti dell'ambiente di test è un compito eseguito come parte della progettazione dei test, quindi è probabile che sia un compito svolto dal ruolo del testing</p> <p>c) Non è corretta. Il ruolo del testing è principalmente responsabile degli aspetti tecnici del testing, come l'analisi dei test, la progettazione dei test, l'implementazione dei test e l'esecuzione dei test. Valutare la testabilità di un oggetto di test è un compito svolto come parte dell'analisi dei test, quindi è probabile che sia un compito svolto dal ruolo del testing</p> <p>d) È corretta. Il ruolo di test management è principalmente responsabile delle attività relative alla pianificazione dei test, al monitoraggio e controllo dei test e al completamento dei test. Pertanto, creare il test completion report, che è l'output principale dell'attività di completamento dei test, è probabile che sia un compito svolto dal ruolo di test management</p>	FL-1.4.5	K2	1
---	---	---	----------	----	---

7	a	<p>a) È corretta. L'approccio whole-team promuove una solida comunicazione e collaborazione tra i membri del team</p> <p>b) Non è corretta. Sebbene l'approccio whole-team prioritizzi la responsabilità collettiva rispetto alla qualità, ogni singolo membro del team è comunque ugualmente responsabile rispetto alla qualità</p> <p>c) Non è corretta. L'approccio whole-team è relativo al modo in cui il team lavora insieme, con l'obiettivo di ottenere rilasci di qualità migliore, ma non si traduce necessariamente in un rilascio più veloce verso gli utenti finali</p> <p>d) Non è corretta. Quando si utilizza l'approccio whole-team, i tester lavorano con i rappresentanti di business per creare i test di accettazione. Non esiste alcun suggerimento che l'approccio riduca la collaborazione con gli utenti di business esterni</p>	FL-1.5.2	K1	1
---	---	---	----------	----	---

8	b	<p>Si consideri ognuno dei vantaggi e degli svantaggi specificati dell'indipendenza del testing:</p> <ul style="list-style-type: none"> i. Idealmente, è richiesta una stretta collaborazione tra tester e sviluppatori, che non aumenta con l'isolamento. Quindi, questo è uno svantaggio ii. I tester e gli sviluppatori hanno background, punti di vista tecnici e potenziali bias (pregiudizi) differenti, permettendo ai tester di mettere in discussione in modo utile le assunzioni fatte dagli stakeholder durante la specifica e l'implementazione del sistema. Quindi, questo è un vantaggio iii. Il principale svantaggio dell'indipendenza del testing è che i tester possono diventare isolati dal team di sviluppo, causando problemi di comunicazione, mancanza di collaborazione e potenzialmente un rapporto conflittuale con i tester, che vengono incolpati di ritardi e di essere colli di bottiglia nel processo di rilascio. Quindi, questo è uno svantaggio iv. Uno degli svantaggi dell'indipendenza del testing è che i tester possono diventare isolati dal team di sviluppo, facendo sì che gli sviluppatori si sentano meno responsabili della qualità. Quindi, questo è uno svantaggio v. Il vantaggio principale dell'indipendenza del testing è che i tester hanno maggiori probabilità di identificare diversi tipi di failure e di difetti rispetto agli sviluppatori, a causa dei loro differenti background, punti di vista tecnici e potenziali bias, compresi i bias cognitivi. Quindi, questo è un vantaggio <p>Quindi: a) Non è corretta</p>	FL-1.5.3	K2	1
---	---	---	----------	----	---

		<p>b) È corretta. Le voci dell'elenco che specificano i benefici sono ii e v</p> <p>c) Non è corretta</p> <p>d) Non è corretta</p>			
9	a	<p>a) È corretta. Ogni livello di test ha obiettivi di test specifici e distinti, poiché ogni livello di test verifica una forma differente dell'oggetto di test (p.e. singolo componente, sistema completo) e la sovrapposizione degli obiettivi di test porterebbe a inutili duplicazioni</p> <p>b) Non è corretta. L'analisi e la progettazione dei test per un determinato livello di test dovrebbero iniziare durante la corrispondente fase di sviluppo, per facilitare l'esecuzione del testing anticipato (p.e. l'analisi e la progettazione dei test di accettazione dovrebbero iniziare durante l'analisi dei requisiti). L'implementazione dei test inizierà generalmente più tardi, e l'esecuzione dei test inizierà durante il livello di test</p> <p>c) Non è corretta. La progettazione dei test per un determinato livello di test dovrebbe iniziare durante la corrispondente fase di sviluppo, per facilitare l'esecuzione del testing anticipato, tuttavia la progettazione dei test (p.e. la generazione dei test case) deve basarsi su una base di test approvata, non su draft iniziali, altrimenti può essere sprecato un effort di test significativo per creare test case per una progettazione che successivamente può cambiare</p> <p>d) Non è corretta. Il quality control si applica a tutte le attività di sviluppo, e questo significa che ogni attività di sviluppo software ha una corrispondente attività di test. Tuttavia, la stessa simmetria non si applica al testing dinamico e statico. Esistono alcune attività del testing statico (p.e. l'analisi statica) per le quali non esiste un'attività del testing dinamico corrispondente</p>	FL-2.1.2	K1	1

10	a	<p>a) È corretta. Il Behavior-Driven Development (BDD) è un esempio conosciuto di approccio test-first allo sviluppo</p> <p>b) Non è corretta. Il Test Level Driven Development non è un esempio corretto di approccio test-first allo sviluppo</p> <p>c) Non è corretta. Il Function-Driven Development non è un esempio corretto di approccio test-first allo sviluppo</p> <p>d) Non è corretta. Il Performance-Driven Development non è un esempio corretto di approccio test-first allo sviluppo</p>	FL-2.1.3	K1	1
11	d	<p>a) Non è corretta. DevOps in genere aumenta la visibilità delle caratteristiche di qualità non-funzionali, come le prestazioni e l'affidabilità</p> <p>b) Non è corretta. I processi automatizzati come la continuous integration/continuous delivery (CI/CD) utilizzati in DevOps facilitano la stabilità degli ambienti di test</p> <p>c) Non è corretta. I processi automatizzati come CI/CD utilizzati in DevOps riducono generalmente la necessità del testing manuale</p> <p>d) È corretta. L'implementazione DevOps può comportare diversi rischi e sfide, che includono la necessità di definire e impostare la pipeline di delivery, di introdurre e mantenere strumenti di CI/CD, e di stabilire e mantenere la test automation</p>	FL-2.1.4	K2	1

12	b	<p>a) Non è corretta. I vantaggi delle retrospettive includono creare legami nel team e imparare attraverso la condivisione dei problemi, nonché avere una migliore collaborazione tra sviluppatori e tester attraverso la revisione e il miglioramento delle pratiche di lavoro. Riprendere le persone perché un membro del team ritiene non abbia contribuito pienamente al raggiungimento della qualità, come richiesto dall'approccio whole-team, non contribuirà alla collaborazione e a creare legami in questo team</p> <p>b) È corretta. Durante la retrospettiva, il gruppo discute gli aspetti del progetto che hanno avuto successo e che dovrebbero essere mantenuti, nonché le aree che potrebbero essere migliorate e come farlo</p> <p>c) Non è corretta. I vantaggi delle retrospettive si basano sull'aumento dell'efficacia e dell'efficienza attraverso i miglioramenti del processo; non sono un'opportunità per sfogarsi e criticare il management e i clienti. Inoltre, i risultati vengono memorizzati generalmente nel test completion report, quindi tutto quello che viene detto durante il meeting potrebbe essere letto da altri stakeholder</p> <p>d) Non è corretta. Le retrospettive sono meeting svolti generalmente alla fine di un'iterazione, in cui i membri del team si focalizzano sulla discussione dei problemi relativi alla qualità che si sono verificati nell'iterazione corrente. Non vengono utilizzate per fare piani o prendere decisioni tecniche per l'iterazione successiva; questo verrebbe fatto nel meeting di pianificazione dell'iterazione all'inizio dell'iterazione successiva</p>	FL-2.1.6	K2	1
----	---	---	----------	----	---

13	a	<p>a) È corretta. Verificare che la funzione di ordinamento inserisca gli elementi della lista o della matrice in ordine crescente significa valutare la correttezza funzionale della funzione di ordinamento, che è parte del testing funzionale</p> <p>b) Non è corretta. Verificare se la funzione di ordinamento soddisfa il requisito non-funzionale di completare l'ordinamento entro un secondo è una forma di testing di efficienza delle prestazioni, che è parte del testing non-funzionale</p> <p>c) Non è corretta. Verificare la facilità con cui la funzione di ordinamento può essere modificata da ordinamento crescente a ordinamento decrescente significa valutare la sua modificabilità, una forma di testing di manutenibilità non-funzionale che è parte del testing non-funzionale</p> <p>d) Non è corretta. Verificare che la funzione di ordinamento continui a funzionare correttamente quando si passa da un'architettura a 32 bit a una a 64 bit significa valutare la sua adattabilità, una forma di testing di portabilità che è parte del testing non-funzionale</p>	FL-2.2.2	K2	1
-----------	---	---	----------	----	---

14	b	<p>a) Non è corretta. Assumere che i tester potrebbero verificare la facilità di modificare il sistema di cambio valuta, significa che verrebbe eseguito il testing di manutenibilità piuttosto che il testing di manutenzione, quindi questo non è un trigger per il testing di manutenzione</p> <p>b) È corretta. Una modifica del sistema (come una correzione o un miglioramento) è un esempio di trigger per il testing di manutenzione. L'eliminazione dell'opzione di rimborso del sistema di cambio valuta è una correzione che porterebbe al testing di manutenzione</p> <p>c) Non è corretta. Se il team Agile ha iniziato a sviluppare una user story che aggiunge una nuova funzionalità di fidelizzazione dei clienti al sistema di cambio valuta, questo porterà al testing della nuova funzionalità e successivamente al regression testing. In questa situazione non è richiesto il testing di manutenzione</p> <p>d) Non è corretta. La riconfigurazione del sistema di cambio valuta per supportare transazioni di valuta sia in lingua locale sia in lingua inglese non è una modifica del sistema, una modifica dell'ambiente operativo o il ritiro di un sistema, che sono i tre trigger per il testing di manutenzione</p>	FL-2.3.1	K2	1
----	---	--	----------	----	---

15	c	<p>a) Non è corretta. La maggior parte dei prodotti di lavoro può essere esaminata utilizzando una forma di testing statico, e un contratto deve essere interpretabile da esseri umani e quindi potrebbe essere sottoposto a review, che è una forma di testing statico</p> <p>b) Non è corretta. La maggior parte dei prodotti di lavoro può essere esaminata utilizzando una forma di testing statico, e un test plan deve essere interpretabile da esseri umani e quindi può essere sottoposto a review, che è una forma di testing statico</p> <p>c) È corretta. La maggior parte dei prodotti di lavoro può essere esaminata utilizzando una forma di testing statico; tuttavia, il testing statico non è adatto per prodotti di lavoro che sono troppo complessi per l'interpretazione umana e che non dovrebbero essere analizzati da strumenti. Il codice encrypted è troppo complesso per gli esseri umani e se è correttamente encrypted non sarà analizzabile dalla maggior parte degli strumenti</p> <p>d) Non è corretta. La maggior parte dei prodotti di lavoro può essere esaminata utilizzando una forma di testing statico, e un test charter deve essere interpretabile da esseri umani e quindi potrebbe essere sottoposta a review, che è una forma di testing statico</p>	FL-3.1.1	K1	1
----	---	---	----------	----	---

16	c	<p>Alcuni tipi di difetti possono essere rilevati solo dal testing statico, come codice irraggiungibile, pattern di progettazioni non implementati come richiesto e difetti in prodotti di lavoro non eseguibili.</p> <p>Alcuni tipi di difetti possono essere rilevati sia dal testing statico sia dal testing dinamico, come un difetto di programmazione che può essere osservato da un reviewer in una code review e che causa un failure osservabile durante il testing dinamico.</p> <p>Alcuni tipi di difetti possono essere rilevati solo dal testing dinamico, come problemi di prestazione o problemi di memoria che possono essere osservati solo durante l'esecuzione del codice o del sistema.</p> <p>Quindi:</p> <ul style="list-style-type: none">a) Non è corretta.b) Non è corretta.c) È corretta.d) Non è corretta.	FL-3.1.2	K2	1
----	---	--	----------	----	---

17	b	<p>Le cinque descrizioni elencate e le corrispondenti attività del processo di review sono:</p> <ol style="list-style-type: none">1. Descrive una parte dell'attività di "comunicazione e analisi"2. Descrive una parte dell'attività di "correzione e reporting"3. Descrive una parte dell'attività di "review individuale"4. Descrive una parte dell'attività di "pianificazione"5. Descrive una parte dell'attività di "inizio della review" <p>Il processo di review generico descritto nello standard ISO/IEC 20246 e che è descritto nel Syllabus, comprende le seguenti attività in questo ordine logico:</p> <ul style="list-style-type: none">• Pianificazione (4)• Inizio della review (5)• Review individuale (3)• Comunicazione e analisi (1)• Correzione e reporting (2) <p>Quindi:</p> <ol style="list-style-type: none">a) Non è correttab) È corretta. La sequenza corretta delle attività è: 4 - 5 - 3 - 1 - 2c) Non è correttad) Non è corretta	FL-3.2.2	K2	1
----	---	--	----------	----	---

18	b	<p>a) Non è corretta. Il manager è responsabile di decidere cosa deve essere sottoposto a review e di allocare le risorse, ad esempio tempo e persone, per la review</p> <p>b) È corretta. Il moderatore (o facilitatore) è responsabile di garantire che i review meeting si svolgano in modo efficace, includendo la gestione del tempo, la mediazione delle discussioni e la creazione di un ambiente “safe” in cui ognuno possa esprimere liberamente le proprie opinioni</p> <p>c) Non è corretta. Il presidente non è un ruolo riconosciuto nelle review</p> <p>d) Non è corretta. Il review leader è responsabile della supervisione del processo di review, come selezionare i membri del team di review, schedulare i review meeting e garantire che la review sia completata con successo</p>	FL-3.2.3	K1	1
19	b	<p>a) Non è corretta. Il documento non si riferisce alla struttura interna dell'oggetto di test, ma specifica il comportamento desiderato dell'oggetto di test. Quindi, le tecniche di test white-box non saranno utili nella progettazione dei test case</p> <p>b) È corretta. Il documento è un requisito che specifica il comportamento desiderato dell'oggetto di test. Quindi, in questo caso le tecniche di test più adeguate sono le tecniche di test black-box (p.e. analisi ai valori limite o testing della tabella delle decisioni)</p> <p>c) Non è corretta. Sebbene sia possibile utilizzare tecniche di test basate sull'esperienza per progettare test case basati su questo documento, le tecniche di test black-box saranno più adeguate. Il documento descrive una regola di business precisa e inoltre, una frase come "supera i \$100" suggerisce l'esistenza di limiti delle partizioni di equivalenza che dovrebbero essere testati utilizzando tecniche di test black-box quali l'analisi ai valori limite</p> <p>d) Non è corretta. Le tecniche di test basate sul rischio non sono un tipo di tecnica di test riconosciuta</p>	FL-4.1.1	K2	1

20	b, e	<p>Esistono due partizioni di equivalenza non ancora coperte, che corrispondono a "sconto per studenti" e "sconto per pensionati".</p> <p>a) Non è corretta. $CY - BY = 64$, quindi questi input corrispondono alla partizione già coperta "nessuno sconto"</p> <p>b) È corretta. $CY - BY = 65$, quindi questi input corrispondono a una partizione non ancora coperta ("sconto per pensionati")</p> <p>c) Non è corretta. $CY - BY = -65$, quindi questi input corrispondono alla partizione già coperta "messaggio di errore"</p> <p>d) Non è corretta. $CY - BY = 18$, quindi questi input corrispondono alla partizione già coperta "nessuno sconto"</p> <p>e) È corretta. $CY - BY = 0$, quindi questi input corrispondono a una partizione non ancora coperta ("sconto per studenti")</p>	FL-4.2.1	K3	1
21	c	<p>Esistono tre partizioni di equivalenza: $\{..., -2, -1\}$, $\{0, 1, 2\}$, $\{3, 4, ...\}$. Per l'analisi ai valori limite a due valori devono essere coperti tutti i valori limite di tutte le partizioni di equivalenza. I valori limite sono -1 (per la partizione "temperatura troppo bassa"), 0, 2 (per la partizione "temperatura OK") e 3 (per la partizione "temperatura troppo alta").</p> <p>Quindi:</p> <p>a) Non è corretta</p> <p>b) Non è corretta</p> <p>c) È corretta. L'opzione corretta è: -1, 0, 2, 3</p> <p>d) Non è corretta</p>	FL-4.2.2	K3	1

22	a	<p>I test case TC1, TC2, TC3 e TC4 coprono, rispettivamente, le regole R2, R3, R7 e R6 della tabella delle decisioni.</p> <p>a) È corretta. Le condizioni "66 anni", "non registrato" e "nessuna esperienza" corrispondono alla regola R4, che non è coperta dai test case esistenti, quindi l'aggiunta di questo test case aumenterà la copertura della tabella delle decisioni</p> <p>b) Non è corretta. Le condizioni "55 anni", "non registrata" e "2 anni di esperienza" corrispondono alla regola R2, già coperta da TC1. Pertanto, l'aggiunta di questo test case non aumenterà la copertura della tabella delle decisioni</p> <p>c) Non è corretta. Le condizioni "19 anni", "registrata" e "5 anni di esperienza" corrispondono alla regola R6, già coperta da TC4. Pertanto, l'aggiunta di questo test case non aumenterà la copertura della tabella delle decisioni</p> <p>d) Non è corretta. I test case esistenti coprono solo 4 delle 7 colonne della tabella delle decisioni. La copertura può essere aumentata aggiungendo test case che coprano le colonne non ancora coperte R1, R4 e R5</p>	FL-4.2.3	K3	1
-----------	---	--	----------	----	---

23	b	<p>a) Non è corretta. Questa sequenza di cinque eventi copre 4 differenti transizioni valide (entrambi gli eventi "Non disponibile" corrispondono alla stessa transizione da S1 a S3). Questo test case copre 4 delle 7 transizioni valide</p> <p>b) È corretta. Questa sequenza di cinque eventi copre 5 differenti transizioni (il primo evento "Disponibile" corrisponde alla transizione da S1 a S2 e il secondo evento "Disponibile" corrisponde alla transizione da S3 a S2, che corrispondono a due transizioni differenti). Questo test case copre 5 delle 7 transizioni valide e raggiunge la copertura più alta delle transizioni valide</p> <p>c) Non è corretta. Questa sequenza di cinque eventi copre 3 differenti transizioni (entrambi gli eventi "Disponibile" corrispondono alla stessa transizione da S1 a S2; entrambi gli eventi "Cambio Camera" corrispondono alla stessa transizione da S2 a S1). Questo test case copre 3 delle 7 transizioni valide</p> <p>d) Non è corretta. Questa sequenza di cinque eventi non rappresenta un test case fattibile, poiché dopo l'evento "Annullamento" il sistema raggiunge lo stato "Fine" e non è possibile eseguire ulteriori transizioni valide</p>	FL-4.2.4	K3	1
----	---	--	----------	----	---

24	c	<p>a) Non è corretta. Una linea di codice con un difetto, quando viene eseguita, non deve necessariamente causare un failure. Per esempio, una linea $x := y / z$ causerà un failure <i>solo</i> quando z è uguale a 0</p> <p>b) Non è corretta. Il 100% di copertura delle istruzioni non garantisce il 100% di copertura dei rami. Ad esempio, un test case con $x=0$ per il codice</p> <pre>1. IF (x=0) THEN 2. A; 3. ENDIF</pre> <p>raggiunge il 100% di copertura delle istruzioni, ma non copre il ramo senza istruzioni ELSE che porta direttamente da 1 a 3</p> <p>c) È corretta. Il 100% di copertura delle istruzioni significa che ogni istruzione eseguibile è stata eseguita almeno una volta</p> <p>d) Non è corretta. Il test case eliminato può raggiungere la copertura di alcune istruzioni che non sono coperte dagli altri due test case, quindi questi due test case rimanenti non raggiungeranno il 100% di copertura delle istruzioni</p>	FL-4.3.1	K2	1
----	---	---	----------	----	---

25	a	<p>a) È corretta. Un punto di forza fondamentale che accomuna tutte le tecniche di test white-box è che durante il testing viene presa in considerazione l'implementazione completa del software, facilitando la rilevazione dei difetti anche quando le specifiche del software sono vaghe, obsolete o incomplete. Questo significa che il testing white-box può rilevare difetti come la codifica (accidentale o deliberata) di una funzionalità extra che non dovrebbe essere presente, che il testing black-box non è in grado di rilevare</p> <p>b) Non è corretta. Il fatto che la copertura possa essere definita con precisione non è la giusta motivazione. Il livello di copertura raggiunto avrebbe un impatto molto maggiore rispetto alla possibilità di misurare la copertura</p> <p>c) Non è corretta. Se il software non implementa uno o più requisiti, è improbabile che il testing white-box rilevi i difetti di omissione che ne derivano</p> <p>d) Non è corretta. Anche se è vero, non è la risposta giusta, perché non esiste alcuna correlazione tra la capacità di essere utilizzato sia nel testing statico sia nel testing dinamico, e l'affermazione che il testing white-box facilita la rilevazione dei difetti quando le specifiche sono vaghe</p>	FL-4.3.3	K2	1
----	---	---	----------	----	---

26	c	La tecnica error guessing anticipa gli errori, i difetti e i failure sulla base della conoscenza del tester. a) Non è corretta. Questo è un esempio di anticipazione dell'errore dello sviluppatore b) Non è corretta. Questo è un esempio di anticipazione del difetto c) È corretta. Questo è un esempio di potenziale root cause di un difetto, che non è né un errore, né un difetto, né un failure, e che è difficile anticipare per un tester d) Non è corretta. Questo è un esempio di anticipazione di un failure, forse basato sull'esperienza di sistemi precedenti in questo dominio applicativo	FL-4.4.1	K2	1
----	---	---	----------	----	---

27	d	<p>a) Non è corretta. Nel testing esplorativo, i test case vengono generalmente creati durante la sessione di testing esplorativo, insieme all'analisi dei test, all'implementazione dei test e all'esecuzione dei test</p> <p>b) Non è corretta. Nel testing esplorativo, i test vengono progettati, eseguiti e valutati contemporaneamente, mentre il tester impara a conoscere l'oggetto di test</p> <p>c) Non è corretta. I risultati del testing esplorativo dipendono fortemente dall'esperienza del tester, quindi, anche se i risultati del testing esplorativo possono essere utilizzati per prevedere il rischio e per valutare se ci saranno pochi o troppi difetti (p.e. rispetto alla precedente sessione di testing esplorativo), non sono un buon esempio di modello predittivo dei difetti che possa prevedere il numero di difetti rimanenti</p> <p>d) È corretta. Durante il testing esplorativo, i tester possono utilizzare tutte le tecniche che ritengono utili</p>	FL-4.4.2	K2	1
28	d	<p>a) Non è corretta. Il planning poker può stimare l'effort per una user story che è già stata scritta. Non aiuta a comprendere cosa dovrebbe essere rilasciato</p> <p>b) Non è corretta. Le review non sono una pratica di scrittura collaborativa delle user story</p> <p>c) Non è corretta. La pianificazione dell'iterazione è una pratica relativa al progetto, utilizzata per pianificare il lavoro, non per comprendere cosa deve essere rilasciato</p> <p>d) È corretta. La conversazione spiega come verrà utilizzato il software e spesso permette al team di definire criteri di accettazione significativi, ottenendo così una vision condivisa di quello che dovrebbe essere rilasciato</p>	FL-4.5.1	K2	1

29	a	<p>a) È corretta. Questo test case è correlato ai criteri di accettazione 2 e 3, perché verificano se è possibile impostare l'intervallo di prezzi (criterio di accettazione 2) e se i risultati si aggiornano dinamicamente dopo aver modificato il filtro della fascia di prezzo (criterio di accettazione 3)</p> <p>b) Non è corretta. Questo test case non è correlato a nessuno dei criteri di accettazione. Verifica se il filtro imposta dinamicamente l'intervallo di prezzo minimo e prezzo massimo predefinito, e non che un cliente possa farlo</p> <p>c) Non è corretta. Questo test case non è correlato a nessuno dei criteri di accettazione. Verifica la funzionalità di cambio valuta, che non è definita in questa user story</p> <p>d) Non è corretta. Questo test case non è correlato a nessuno dei criteri di accettazione. Verifica la compatibilità dell'applicazione con diversi browser, che non viene definita in questa user story</p>	FL-4.5.3	K3	1
30	b, d	<p>a) Non è corretta. L'approvazione del budget è un esempio di criterio di ingresso. Non avrebbe senso approvare il budget per un'attività che è già stata eseguita</p> <p>b) È corretta. L'esaurimento del budget può essere considerato un valido criterio di uscita</p> <p>c) Non è corretta. La disponibilità di risorse è un esempio di criterio di ingresso per il testing</p> <p>d) È corretta. La copertura è una misura di completezza, quindi è un tipico criterio di uscita</p> <p>e) Non è corretta. Questo è un esempio di criterio di ingresso, verificato prima che il progetto inizi</p>	FL-5.1.3	K2	1

31	a	<p>Utilizzando la tecnica three-point estimation, la stima finale (E) è calcolata come: $E = (a + 4*m + b) / 6,$ dove a è la stima più ottimistica, m è la stima più probabile e b è la stima più pessimistica.</p> <p>Quindi:</p> <p>a) È corretta. In questo caso, la stima per l'esecuzione di un singolo test case è: $E = (1h + 4*3h + 8h) / 6 = 3,5 \text{ ore}$ Quindi, il tempo totale necessario al tester per eseguire 4 test case è: $3,5h * 4 = 14 \text{ ore}$</p> <p>b) Non è corretta. Si veda spiegazione a) c) Non è corretta. Si veda spiegazione a) d) Non è corretta. Si veda spiegazione a)</p>	FL-5.1.4	K3	1
32	b	<p>TC1 raggiunge la copertura più alta (4/7 perché copre Req1, Req3, Req4 e Req7), quindi dovrebbe essere eseguito per primo. Req2, Req5 e Req6 non sono ancora coperti. Il test case successivo che raggiunge la copertura aggiuntiva più alta dei requisiti rimanenti è TC3, che copre 2 dei 3 requisiti (Req5 e Req6). Quindi, TC3 dovrebbe essere eseguito come secondo. L'unico requisito non ancora coperto è Req2, che è coperto da TC4. Di conseguenza, TC4 dovrebbe essere eseguito come terzo test case. Infine, l'ultimo test case eseguito sarà TC2.</p> <p>Quindi:</p> <p>a) Non è corretta b) È corretta c) Non è corretta d) Non è corretta</p>	FL-5.1.5	K3	1

33	c	<p>a) Non è corretta. I quadranti del testing non hanno nulla a che fare con la descrizione delle relazioni tra i livelli di test</p> <p>b) Non è corretta. I quadranti del testing non possono aiutare a valutare alcun tipo di copertura</p> <p>c) È corretta. I quadranti del testing consentono ai manager e agli altri stakeholder di comprendere le relazioni tra i tipi di test, le attività che supportano (supportare il team o criticare il prodotto) e il punto di vista su cui si focalizzano (orientato al business o alla tecnologia)</p> <p>d) Non è corretta. I quadranti del testing non sono un modello psicologico</p>	FL-5.1.7	K2	1
34	b	<p>La valutazione del rischio può utilizzare un approccio quantitativo o qualitativo, o un mix di entrambi. Nell'approccio quantitativo il livello di rischio viene calcolato come prodotto tra la probabilità del rischio e l'impatto del rischio. Quindi, livello di rischio = probabilità del rischio * impatto del rischio</p> <p>Quindi, impatto del rischio = livello di rischio / probabilità del rischio.</p> <p>Nel nostro caso, impatto del rischio = \$1.000 / 50% = \$1.000 / 0,5 = \$2.000.</p> <p>Quindi:</p> <p>a) Non è corretta</p> <p>b) È corretta</p> <p>c) Non è corretta</p> <p>d) Non è corretta</p>	FL-5.2.1	K1	1
35	b, e	<p>a) Non è corretta. Lo scope creep è un esempio di rischio di progetto correlato a problemi tecnici</p> <p>b) È corretta. Un'architettura scadente è un esempio di rischio di prodotto, poiché si riferisce a una caratteristica del prodotto</p> <p>c) Non è corretta. La riduzione dei costi è un esempio di rischio di progetto, correlato a problemi organizzativi</p> <p>d) Non è corretta. Lo scarso supporto degli strumenti è un esempio di rischio di progetto correlato a problemi tecnici</p> <p>e) È corretta. Il tempo di risposta troppo lungo è un esempio di rischio di prodotto, poiché si riferisce a una caratteristica del prodotto</p>	FL-5.2.2	K2	1

36	c	a) Non è corretta. Questo è un esempio di supporto al continuo controllo del testing, ed è uno degli obiettivi dei test report b) Non è corretta. Questo è un esempio di riepilogo delle attività di test eseguite a un determinato livello di test., ed è uno degli obiettivi dei test report c) È corretta. Questo è lo scopo di un defect report, non di un test report d) Non è corretta. Questo è uno degli obiettivi dei test report	FL-5.3.2	K2	1
----	---	---	----------	----	---

37	d	<p>a) Non è corretta. Il risk management consiste nell'analisi del rischio e nel controllo del rischio. Nessuna di queste attività supporta l'identificazione dei file che compongono il rilascio, perché queste attività si occupano dei rischi, non degli elementi di configurazione</p> <p>b) Non è corretta. Il monitoraggio dei test si occupa di raccogliere informazioni sul testing. Queste informazioni vengono utilizzate per valutare l'avanzamento dei test e per misurare se i criteri di uscita dei test o le attività di test associate ai criteri di uscita sono stati soddisfatti, come il raggiungimento degli obiettivi di copertura dei rischi di prodotto, dei requisiti o di altri criteri di accettazione. Il controllo dei test utilizza le informazioni del monitoraggio dei test per fornire, sotto forma di direttive di controllo, una guida e le azioni correttive necessarie per ottenere il testing più efficace ed efficiente. Nessuna di queste attività si occupa della gestione degli elementi di configurazione</p> <p>c) Non è corretta. L'approccio whole-team si basa sulla capacità del tester di lavorare efficacemente in un contesto di team e di contribuire positivamente agli obiettivi del team. Pertanto, si focalizza sui problemi relativi al team, non sugli elementi di configurazione</p> <p>d) È corretta. Il configuration management fornisce una disciplina per identificare, controllare e tracciare i prodotti di lavoro. Il configuration management mantiene un record degli elementi di configurazione modificati quando viene creata una nuova baseline. Utilizzando il configuration management, è possibile ritornare a una baseline precedente per riprodurre i risultati dei test precedenti</p>	FL-5.4.1	K2	1
----	---	---	----------	----	---

38	a	<p>a) È corretta. Aggiungere queste informazioni consente allo sviluppatore di utilizzare gli stessi dati di input, così è più probabile che saranno in grado di riprodurre rapidamente il failure e quindi di identificare più velocemente il difetto</p> <p>b) Non è corretta. Aggiungere il valore della Priorità non aiuterà a riprodurre il difetto stesso</p> <p>c) Non è corretta. Sebbene alcune di queste informazioni possano essere utili, l'aggiunta dei memory dump e dei database snapshot dopo ogni passo sarà troppo, perché la maggior parte di questi artefatti conterrà informazioni inutili per lo sviluppatore, e renderà il report meno leggibile. Inoltre, lo sviluppatore dovrà dedicare molto tempo all'analisi di queste informazioni, allungando così il processo di correzione</p> <p>d) Non è corretta. La domanda riguardava la possibilità di aiutare lo sviluppatore a riprodurre il difetto osservato per una specifica configurazione dell'ambiente</p>	FL-5.5.1	K3	1
-----------	---	---	----------	----	---

39	b	<p>Si considerino le categorie di strumenti elencate:</p> <ul style="list-style-type: none"> i. Strumenti di collaborazione - facilitano la comunicazione ma non facilitano l'esecuzione dei test. ii. Strumenti DevOps - supportano la pipeline di delivery DevOps, il tracciamento del flusso di lavoro, i processi di creazione automatizzati e la CI/CD. La pipeline di delivery e la CI/CD facilitano l'esecuzione dei test, ad esempio il testing di componente per la CI. iii. Strumenti di management - aumentano l'efficienza del processo di test facilitando la gestione del ciclo di vita dello sviluppo software, dei requisiti, dei test, dei difetti e della configurazione. Ma non facilitano l'esecuzione dei test. iv. Strumenti di testing non-funzionale - permettono al tester di eseguire testing non-funzionale difficile o impossibile da eseguire manualmente. Il testing non-funzionale può includere sia il testing statico sia il testing dinamico, includendo l'esecuzione dei test. v. Strumenti di progettazione e implementazione dei test - facilitano la generazione di test case, dati di test e procedure di test, ma non facilitano l'esecuzione dei test. <p>Quindi:</p> <ul style="list-style-type: none"> a) Non è corretta b) È corretta. Sia gli strumenti DevOps (ii) sia gli strumenti di testing non-funzionale (iv) facilitano l'esecuzione dei test c) Non è corretta d) Non è corretta 	FL-6.1.1	K2	1
----	---	---	----------	----	---

40	c	a) Non è corretta. Questo sarebbe un vantaggio della test automation, piuttosto che un rischio b) Non è corretta. Questo è normalmente considerato un vantaggio della test automation c) È corretta. Se la test automation è incompatibile con la piattaforma di sviluppo non sarà possibile integrarli. Non sarà per esempio possibile passare gli input del test all'oggetto di test e di ricevere i risultati dei test dall'oggetto di test d) Non è corretta. Questo è normalmente considerato un vantaggio della test automation	FL-6.2.1	K1	1
----	---	--	----------	----	---