

## Test 06 – Testing

**Un Boundary use case describe (indicare la/e risposta/e corrette):**

- la configurazione del sistema
- le scelte relative allo startup, allo shutdown
- la gestione delle condizioni limite.

**A cosa corrisponde la seguente definizione?: La misura di successo con cui il comportamento osservato di un sistema è conforme ad una certa specifica del relativo comportamento.**

- Correttezza
- Affidabilità
- Conformità

**Che cosa è un fallimento?**

- Qualsiasi deviazione del comportamento osservato dal comportamento specificato.
- Quando il sistema è in uno stato tale che ogni ulteriore elaborazione da parte del sistema porta ad un errore.
- La causa meccanica o algoritmica di un errore

**Quale/i delle seguenti affermazioni è/sono corretta/e?**

- Un programma è esercitato da un caso di test (insieme di dati di input)
- Un test è formato da un caso di test
- L'esecuzione del test consiste nell'esecuzione del programma per tutti i casi di test
- Un test ha successo se non rileva malfunzionamenti del programma

**Quale delle seguenti affermazioni vale per il Fault Avoidance?**

- Tecniche che tentano di correggere i difetti del sistema software
- Tecniche che usano metodologie di programmazione per ridurre la complessità
- Tecniche che intendono individuare i difetti eseguendo il sistema software

**Quale delle seguenti affermazioni vale per il Fault Tolerance?**

- Tecniche che assumono che un sistema possa essere realizzato con bug
- Tecniche per ispezionare e recuperare dai fallimenti del sistema
- Tecniche per gestire i fallimenti a compile-time

**Le tecniche Fault detection includono (indicare le risposte corrette):**

- debugging
- decoding
- reviewing

**Durante l'ispezione ai fini del fault detection (indicare le risposte corrette):**

- Lo sviluppatore presenta gli artefatti.
- Si controlla l'efficienza degli algoritmi con le richieste funzionali
- Lo sviluppatore interviene solo se si richiedono chiarimenti

**Quando si effettua il testing di regressione?**

- Dopo aver effettuato il test di integrazione e prima di effettuare il test di sistema
- Dopo aver effettuato il testing di sistema e prima di effettuare il testing di accettazione
- Dopo aver effettuato delle modifiche al codice per correggere un fault

**Il testing che mira a valutare l'aderenza del sistema ai requisiti non funzionali viene denominato**

- Integration testing
- Acceptance testing
- Performance testing

**L'insieme dei dati di input e dei risultati attesi che esercitano un componente software con lo scopo di causare una failure viene denominato:**

- Test Plan
- Test Case
- Test Incident

**Un test case ha (indicare le risposte corrette)::**

- 5 attributi
- 4 attributi
- 6 attributi

**Quale di queste affermazioni è vera:**

- Il testing white-box consente di capire se non ho implementato delle funzionalità
- Il testing black-box consente di individuare caratteristiche non richieste
- Il testing black-box consente di capire se non ho implementato delle funzionalità

**Quale di queste affermazioni è vera?**

- con il top down integration testing ho bisogno di driver
- con il bottom up integration testing ho bisogno di driver
- con il sandwich testing non ho bisogno di driver

**Quale di queste affermazioni è vera?**

- Un test stub è una implementazione completa di componenti da cui la componente testata dipende
- Un test driver è una implementazione parziale di una componente che dipende dalla componente testata (componente che chiama la componente testata).
- Un test driver è una implementazione parziale di una componente da cui la componente testata dipende

**Quale di queste affermazioni è vera?**

- Il system testing si focalizza solo sulla verifica dei requisiti funzionali del sistema da testare
- L'Usability testing si preoccupa di trovare le differenze tra l'interfaccia grafica del sistema sviluppato e quella attesa dall'utente
- Per Component inspection si intende il trovare i fault in una componente individuale attraverso l'ispezione manuale del codice sorgente

**Quando si usa uno Scenario test (Usability testing) (indicare (indicare le risposte corrette):**

- Si utilizzano mock-up o di storyboard
- Gli utenti possono interagire direttamente con il sistema
- Viene presentato un uso concreto del sistema

**Quale di queste affermazioni è vera? Usando Unit testing**

- Aumenta la complessità testando più unità in parallelo
- È più facile correggere i bug, poiché poche componenti sono coinvolte
- Diverse unità non possono essere testate in parallelo

**Relativamente al Pilot Testing, quale delle seguenti affermazioni è falsa**

- Il sistema è installato e usato da un insieme di utenti selezionati
- Linee guida o scenari sono forniti agli utenti
- Sistemi pilota sono utili quando un sistema è costruito senza un insieme di richieste specifiche, o senza un particolare cliente in mente

**Dall'informazione della matrice di tracciabilità in cui ho collegato i requisiti ai relativi test case**

- posso ricavare quali requisiti non sono stati realizzati
- posso ricavare quali requisiti sono testati
- posso determinare se ho realizzato un white box testing

**Relativamente al Top-Down Integration Testing, quale delle seguenti affermazioni è falsa:**

- I test cases possono essere definiti in termine delle funzionalità del sistema
- I test cases si possono riutilizzare nelle varie iterazioni
- Gli stub non devono consentire tutte le condizioni da testare

**Descrivere nel dettaglio l'equivalence class testing, specificando in particolare: obiettivo**

---

---

---

**quali caratteristiche devono soddisfare le classi di equivalenza,**

---

---

---

**specificare che tipo di test è (whitebox o blackbox, perché?),**

---

---

**quali sono i diversi tipi di equivalence class-testing**

---

---

**Descrivere il category partition ed elencare i passi della strategia**

---

---

---

---