

Familienname, Vorname: _____

Firmenadresse: _____

Telefon: _____

Fax : _____

E-Mail-Adresse: _____

Rechnungsadresse: _____

Schulungsunternehmen: _____

Referent: _____

Foundation Level Probepfprüfung
CTFL Syllabus Version 2011 deutschsprachig V.1.0.1
ISTQB Glossar V.3
Version 2.4

ISTQB® Certified Tester Foundation Level

(Hinweis: Wenn nicht anders gekennzeichnet ist nur eine der vorgegebenen Antworten jeweils zutreffend.)

Bearbeitungszeit: 60 Minuten

Einführung

Dies ist eine Probepfprüfung. Sie hilft den Kandidaten bei ihrer Vorbereitung auf die Zertifizierungsprüfung. Enthalten sind Fragen, deren Format der regulären ISTQB®¹ / STB/ GTB² Certified Tester Foundation Level Prüfung ähnelt.

Es ist strengstens verboten, diese Prüfungsfragen in einer echten Prüfung zu verwenden.

- 1) Jede Einzelperson und jeder Schulungsanbieter kann diese Probepfprüfung in einer Schulung verwenden, wenn ISTQB® als Quelle und Copyright-Inhaber der Probepfprüfung anerkannt wird.
- 2) Jede Einzelperson oder Gruppe von Personen kann diese Probepfprüfung als Grundlage für Artikel, Bücher oder andere abgeleitete Schriftstücke verwenden, wenn ISTQB® als Quelle und Copyright-Inhaber der Probepfprüfung bestätigt wird.
- 3) Jedes vom ISTQB® anerkannte nationale Board kann diese Probepfprüfung übersetzen und öffentlich zugänglich machen, wenn ISTQB® als Quelle und Copyright-Inhaber der Probepfprüfung bestätigt wird.
- 4) Zu fast jeder Frage wird genau eine zutreffende Lösung erwartet. Bei den Ausnahmen wird explizit auf die Möglichkeit mehrerer Antworten hingewiesen.

Allgemeine Angaben zur Probepfprüfung:

Anzahl der Fragen: 40

Dauer der Prüfung: 60 Minuten

Gesamtpunktzahl: 40

Punktzahl zum Bestehen der Prüfung: 26 (oder mehr)

Prozentsatz zum Bestehen der Prüfung: 65% (oder mehr)

¹ International Software Testing Qualifications Board

² German Testing Board & Austrian Testing Board & Swiss Testing Board (deutschsprachig)

Fragen zum Thema
”Grundlagen des Testens”

1. Welche der folgenden Aussagen beschreibt AM BESTEN einen der sieben Grundsätze des Software-Testens? [K2]

a)	Automatisierte Tests sind besser als manuelle Tests zum Vermeiden von „Vollständigem Testen“ geeignet.	<input type="checkbox"/>
b)	Vollständiges Testen ist - mit genügend Anstrengung und Werkzeugunterstützung - für alle Arten von Software praktikabel.	<input type="checkbox"/>
c)	Es ist normalerweise unmöglich, alle Input / Output-Kombinationen für ein Software-System zu testen.	<input type="checkbox"/>
d)	Der Zweck des Testens ist es, die Abwesenheit von Fehlern nachzuweisen.	<input type="checkbox"/>

Begründung

- a) FALSCH – Erschöpfendes Testen ist unmöglich. Egal, ob manuell oder automatisiert. (Abschn. 1.3, Grundsatz 2).
- b) FALSCH – Erschöpfendes Testen ist unmöglich, egal wieviel Aufwand wir in den Test investieren. (Abschn. 1.3, Grundsatz 2).
- c) KORREKT – Grundsatz 2 (Abschn. 1.3) besagt: “Ein vollständiger Test, bei dem alle möglichen Eingabewerte und deren Kombinationen ... ausgeführt werden, ist nicht möglich, mit Ausnahme von sehr trivialen Testobjekten.”
- d) FALSCH – Diese Aussage widerspricht dem Grundsatz 1 (Abschn. 1.3): “Mit Testen wird das Vorhandensein von Fehlerzuständen nachgewiesen. Mit Testen lässt sich nicht beweisen, dass keine Fehlerzustände im Testobjekt vorhanden sind!

2. Welche der folgenden Aussagen beschreibt ZUTREFFEND ein Ziel für ein Testteam? [K1]

a)	Es soll ermittelt werden, ob genügend Komponententests im Systemtest ausgeführt wurden.	<input type="checkbox"/>
b)	Es sollen so viele Fehlerwirkungen wie möglich aufgedeckt werden, so dass die Fehlerursachen lokalisiert und korrigiert werden können.	<input type="checkbox"/>
c)	Es soll aufgezeigt werden, dass alle möglichen Fehlerzustände identifiziert wurden.	<input type="checkbox"/>
d)	Es soll aufgezeigt werden, dass alle verbleibenden Fehlerzustände keine Fehlerwirkungen verursachen werden.	<input type="checkbox"/>

Begründung

- a) FALSCH – Komponententest ist nicht Teil des Systemtest. (Abschn. 2.2.1 und 2.2.3).
- b) KORREKT – Diese ist die Hauptaufgabe eines Testteams. (Abschn. 1.2, Ziele, 1. dot).
- c) FALSCH – Grundsatz 1 besagt, dass vollständiges Testen unmöglich ist, und es kann nicht bewiesen werden, dass alle Fehlerzustände identifiziert wurden.
- d) FALSCH – Um eine Einschätzung treffen zu können, ob ein Defekt einen Fehler verursacht oder nicht, muss man den Fehlerzustand zunächst erkennen. Zu sagen, dass verbleibende Fehlerzustände keine Fehlerwirkungen verursachen, bedeutet implizit, dass alle Fehlerzustände gefunden wurden. Dies widerspricht erneut Grundsatz 1.

3. Welche der folgenden Aktivitäten ist Teil der Hauptaktivität „Testanalyse und Testentwurf“ im Fundamentalen Testprozess? [K1]

a)	Definition der Testziele	<input type="checkbox"/>
b)	Review der Testbasis	<input type="checkbox"/>
c)	Erstellen von Testsuiten basierend auf dem Testablauf	<input type="checkbox"/>
d)	Analyse der „Lessons learned“ zur Prozessverbesserung	<input type="checkbox"/>

Begründung

- a) FALSCH – diese Aktivität wird in der Hauptaktivität “Testplanung” durchgeführt (Abschnitt 1.4.1, Satz 1).
- b) KORREKT – diese Aktivität wird in der Hauptaktivität “Testanalyse und Design” durchgeführt. (Abschnitt 1.4.2; 1. dot)
- c) FALSCH – diese Aktivität wird in der Hauptaktivität “Testrealisierung und Testdurchführung” durchgeführt (Abschnitt 1.4.3; 3. dot)
- d) FALSCH – diese Aktivität wird in der Hauptaktivität “Abschluss der Testaktivitäten” durchgeführt (Abschnitt 1.4.5); 6. dot).

4. Nachfolgend finden Sie eine Liste von Problemen, die während des Testens oder im Betrieb beobachtet werden können. Welches Problem ist eine Fehlerwirkung? [K2]

a)	Das Produkt stürzte ab, als der Benutzer eine Option in einer Dialogbox auswählte.	<input type="checkbox"/>
b)	Eine kompilierte Quellcodedatei wurde in der falschen Version zum Build hinzugefügt.	<input type="checkbox"/>
c)	Der Berechnungsalgorithmus verwendet die falschen Eingangsvariablen.	<input type="checkbox"/>
d)	Der Entwickler hat die Anforderungen an den Algorithmus falsch interpretiert.	<input type="checkbox"/>

Begründung

- a) KORREKT – Ein Absturz ist vom Anwender deutlich spürbar. Ein Ausfall ist eine äußere Manifestation eines Fehlerzustands, d.h. eine Fehlerwirkung.
- b) FALSCH – Diese Art von Fehlern (Fehlhandlungen) wird nicht unbedingt zu einer sichtbaren oder spürbaren Fehlerwirkung führen; zum Beispiel wenn die Änderungen in der neuen Version der Quelldatei nur in den Kommentaren vorgenommen wurden.
- c) FALSCH – Verwendung von falschen Eingabevariablen wird nicht unbedingt zu einer sichtbaren oder spürbaren Fehlerwirkung führen; zum Beispiel, wenn niemand diesen speziellen Algorithmus verwendet; oder wenn die falsche Eingabevariable einen ähnlichen Wert wie die richtige Eingabevariable hat; oder wenn das FALSCHER Resultat des Algorithmus nicht verwendet wird. "Fehler in Software, Systemen oder Dokumenten können, müssen aber nicht zu einer Fehlerwirkung führen." (Abschnitt 1.1.2, 1. Absatz, letzter Satz).
- d) FALSCH – Diese Art von Fehlern wird nicht notwendigerweise zu einer Fehlerwirkung führen; zum Beispiel, wenn niemand diesen speziellen Algorithmus verwendet.

5. Welche der folgenden Einstellungen, Qualifikationen oder Handlungen fñhren zu Problemen (oder Konflikten) innerhalb von gemischten Tester-Entwickler-Teams, wenn sie in Reviews und Tests beobachtet werden? [K1]

a)	Tester und Entwickler sind neugierig und fokussiert genug, um Fehler zu finden.	<input type="checkbox"/>
b)	Tester und Entwickler sind genñgend qualifiziert, um Fehlerwirkungen und Fehlerzustände zu finden.	<input type="checkbox"/>
c)	Tester und Entwickler kommunizieren Fehler als Kritik an Personen und nicht als Kritik an dem Software-Produkt.	<input type="checkbox"/>
d)	Tester erwarten, dass Fehlerzustände im Software-Produkt existieren, welche von Entwicklern nicht gefunden und behoben wurden.	<input type="checkbox"/>

Begrñndung

- a) FALSCH. Das ist keine Situation, die zu Konflikten fñhrt. Tester und Entwickler sollten fokussiert sein, Fehler zu finden.
- b) FALSCH. Das stimmt, es gibt deswegen keine Probleme.
- c) KORREKT. Laut Lehrplan sollten Entwickler und Tester neutral und sachbezogen kommunizieren und Kritik an verantwortlichen Personen vermeiden (Abschn. 1.5, 9. Absatz, 2. dot), weil das zu Konflikten innerhalb des Testteams fñhrt.
- d) FALSCH. Die Rolle des Testers im Team ist es, Fehler im Softwareprodukt zu finden, die nicht von den Entwicklern gefunden und behoben wurden. Das ist bei sachbezogener Kommunikation kein Problem.

6. Welche der folgenden Aussagen sind WAHR? [K2]

- A. Testen von Software kann erforderlich sein, um gesetzlichen oder vertraglichen Anforderungen gerecht zu werden.**
- B. Testen von Software ist vor allem notwendig, um die Qualitat des Produkts zu verbessern, welches von Entwicklern freigegeben wird.**
- C. Konsequentes Testen und das Beheben gefundener Fehlerzustande kann dazu beitragen, das Risiko auftretender Probleme in einer Produktivumgebung zu reduzieren.**
- D. Konsequentes Testen wird manchmal fur nicht triviale Testobjekte verwendet, um zu beweisen, dass alle Fehlerwirkungen gefunden wurden.**

a)	A, B und C sind wahr; D ist falsch.	<input type="checkbox"/>
b)	A ist wahr; B, C und D sind falsch.	<input type="checkbox"/>
c)	A und C sind wahr, B und D sind falsch.	<input type="checkbox"/>
d)	C und D sind wahr, A und B sind falsch.	<input type="checkbox"/>

Begrundung

- A. WAHR. Softwaretests sind erforderlich, um die vertraglichen und gesetzlichen Anforderungen zu erfullen. (Abschn. 2.2.4, regulatorischer und vertraglicher Abnahmetest).
- B. WAHR. Softwaretests sind vor allem erforderlich um die Qualitat der Arbeit von Entwicklern zu verbessern. (Lernziel LO-1.1.4; Abschn. 1.1.3, 1. Absatz; Abschn. 1.1.4, 2. Absatz; Abschn. 1.5, letzte Aufzahlung, 1. dot)..
- C. WAHR. Eines der Hauptziele ist es das Risiko von Fehlerzustanden (in der Betriebsumgebung) zu reduzieren. (Abschn. 1.2, Ziele 1 und 4).
- D. FALSCH. Es ist nicht moglich zu beweisen, dass keine Fehler vorhanden sind. (Abschn. 1.3, Grundsatz 1), insbesondere nicht fur nicht triviale Testobjekte, bei denen kein „vollstandiger Test“ moglich ist (Abschn. 1.3, Grundsatz 2).

Deshalb gilt:

- a.) **ist korrekt.** weil genau A, B und C wahr sind.
- b) ist falsch, weil B und C nicht falsch (sondern wahr) sind.
- c) ist falsch, weil B nicht falsch (sondern wahr) ist.
- d) ist falsch, weil D nicht wahr (sondern falsch) ist und weil A und B wahr (und nicht falsch) sind.

7. Welche der folgenden Aussagen beschreibt den Unterschied zwischen Testen und Debuggen richtig? [K2]

a)	Testen identifiziert die Ursache von Fehlerwirkungen. Debuggen analysiert die Fehlerzustände und schlägt Präventionsmaßnahmen vor.	<input type="checkbox"/>
b)	Dynamische Tests zeigen Fehlerwirkungen auf, die durch Fehlerzustände verursacht wurden. Debuggen findet, analysiert und beseitigt die Ursachen von Fehlerwirkungen.	<input type="checkbox"/>
c)	Testen entfernt Fehlerzustände. Debugging identifiziert die Ursachen von Fehlerwirkungen.	<input type="checkbox"/>
d)	Dynamische Tests verhindern die Ursache von Fehlerwirkungen. Debugging entfernt die Fehlerwirkungen.	<input type="checkbox"/>

Begründung

- a) FALSCH. Durch Testen können nicht die Ursachen von Fehlerwirkungen identifiziert werden, sondern nur durch Debugging (Abschn. 1.2, 2.letzter Absatz, 3. Satz).
- b) WAHR. Dynamisches Testen zeigt Fehlerwirkungen auf, die durch Fehlerzustände verursacht wurden. Durch Debuggen können die Ursachen von Fehlerwirkungen analysiert und beseitigt werden. (Abschn. 1.2, 2.letzter Absatz).
- c) FALSCH. Durch Testen werden keine Fehlerzustände entfernt, sondern nur durch Debugging (Abschn. 1.2, 2.letzter Absatz, 3. Satz)..
- d) FALSCH. Durch dynamische Tests können die Ursachen von Fehlerwirkungen (d.h. Fehlerzustände) nicht verhindert werden, sondern nur das Vorhandensein von Fehlerzuständen nachgewiesen werden (Abschn. 1.3, Grundsatz 1).

Fragen zum Thema
"Testen im Software-Lebenszyklus"

8. Welche der folgenden Aussagen beschreibt AM BESTEN nicht-funktionales Testen? [K1]

a)	Nichtfunktionales Testen ist der Testprozess eines integrierten Systems, welcher prüft, ob das System die spezifizierten Anforderungen erfüllt.	<input type="checkbox"/>
b)	Nichtfunktionales Testen ist der Testprozess zur Überprüfung der Konformität eines Systems mit den Programmierkonventionen.	<input type="checkbox"/>
c)	Nichtfunktionales Testen ist das Testen ohne Kenntnis der internen Struktur eines Systems.	<input type="checkbox"/>
d)	Nichtfunktionales Testen ist das Testen von Systemmerkmalen wie Benutzbarkeit, Zuverlässigkeit oder Änderbarkeit.	<input type="checkbox"/>

Begründung

- a) FALSCH, dies ist die Definition für einen Systemtest. (Abschn. 2.2.3, Testbasis, 1. dot).
- b) FALSCH, dies ist die Aufgabe der werkzeuggestützten statischen Analyse (Abschn. 3.3, Typische Fehlerzustände, 2. dot, und Abschn. 6.1.4, statische Analysewerkzeuge, 1. Satz).
- c) FALSCH, dies ist die Definition für einen Black-Box Test (Abschn. 4.2, Hintergrund, 3. Absatz („Black-Box-Verfahren ...“)).
- d) KORREKT, nicht-funktionales Testen beinhaltet u.a. das Testen gegen die Qualitätsmerkmale Benutzbarkeit, Zuverlässigkeit und Änderbarkeit. (Abschn. 2.3.2, 1. Satz).

9. Was ist eine wichtige Tatigkeit bei der Arbeit mit Softwareentwicklungsmodellen? [K1]

a)	Falls notig, sollten die Modelle auf Projekt- und Produkteigenschaften angepasst werden.	<input type="checkbox"/>
b)	Das Wasserfall-Modell auswahlen, weil es das bewahrteste Modell ist.	<input type="checkbox"/>
c)	Mit dem V-Modell beginnen und dann entweder auf iterative oder inkrementelle Modelle wechseln.	<input type="checkbox"/>
d)	Die Organisation auf das gewahlte Modell anpassen und nicht umgekehrt.	<input type="checkbox"/>

Begrundung

- a) KORREKT – Modelle stellen generelle Richtlinien bereit – nicht einen Schritt-fur-Schritt Prozess, der zu befolgen ist. (Abschn. 2.1, letzter Absatz).
- b) FALSCH – Das Wasserfallmodell ist nur eines der mogliche Modelle, die von einem Team ausgewahlt werden konnen. Es kommt im Lehrplan nicht vor, wird also erst recht nicht „als bewahrtes Modell“ empfohlen.
- c) FALSCH – Das V-Modell (Abschn. 2.1.1) als sequentielles Modell ist nicht kompatibel mit den iterativ-inkrementellen Modellen (Abschn. 2.1.2).. Deshalb macht die beschriebene Reihenfolge keinen Sinn.
- d) FALSCH – Modelle werden passend zu einer Situation und einem Projekt ausgewahlt und nicht umgekehrt (Abschn. 2.1, letzter Absatz).

10. Welches der folgenden Merkmale ist ein „Merkmal des guten Testens“ und gilt für alle Softwareentwicklungsmodelle? [K1]

a)	Der Abnahmetest ist immer die letzte angewendete Teststufe.	<input type="checkbox"/>
b)	Alle Teststufen sind für jedes einzelne entwickelte Feature geplant und abgeschlossen.	<input type="checkbox"/>
c)	Sobald die erste Komponente ausgeführt werden kann, werden die Tester beteiligt.	<input type="checkbox"/>
d)	Für jede Entwicklungstätigkeit gibt es eine korrespondierende Testaktivität.	<input type="checkbox"/>

Begründung

- a) FALSCH – Dies ist richtig für Projekte mit Entwicklungsmodellen, die einen Akzeptanztest haben. Manche Projekte/Modelle haben diese Teststufe aber nicht (siehe Abschnitt 2.1 des Lehrplans); durch die Ergänzung „immer“ ist klar das die Aussage FALSCH ist.
- b) FALSCH – In einigen Fällen werden Teststufen in Entwicklungsmodellen nicht benötigt; weil es im konkreten Projekt eine Konstellation geben kann in der die Anwendung einer Teststufe keinen Sinn macht. Zum Beispiel: Wenn Komponenten von Lieferanten zugekauft wurden oder als Bibliothek zugekauft wurden, ist für diese Komponenten kein Komponententest durchzuführen.
- c) FALSCH – Tester sollten bereits früher involviert werden bevor die Implementierung abgeschlossen wurde. Zum Beispiel: Tester sollten in das Review der Anforderungsspezifikationen involviert werden. (vgl. Abschn. 1.4.2, 1. dot).
- d) KORREKT – “In jedem Entwicklungslebenszyklus findet man einige Charakteristika für gutes Testen: Zu jeder Entwicklungsaktivität gibt es eine zugehörige Aktivität im Testen.” (Abschnitt 2.1.3), 1. dot)

11. Wozu werden beispielsweise Wartungstests verwendet? [K1]

a)	Zur Fehlerkorrektur während der Entwicklungs- und Debugging-Phase eines neuen Systems.	<input type="checkbox"/>
b)	Für Tests bei Erweiterungen eines produktiv eingesetzten Systems.	<input type="checkbox"/>
c)	Für die Verfolgung von Beschwerden über die Systemqualität während des Anwender-Abnahmetests.	<input type="checkbox"/>
d)	Zur Integration von Funktionen während der Entwicklung eines neuen Systems.	<input type="checkbox"/>

Begründung

- a) FALSCH – Testen eines neuen Systems ist kein „Wartungstest“ (Abschn. 2.4, Absätze 1 bis 3).
- b) KORREKT – Test eines Systems nach einer Änderung (z.B. der Umgebung) wird als „Wartungstest“ bezeichnet (Abschn. 2.4, 1. Absatz, letzter Satz, und 2. Absatz, 1. Satz).
- c) FALSCH – Der Umgang mit Beschwerden während eines Abnahmetest ist kein „Wartungstest“. Der setzt erst nach der Abnahme und Inbetriebnahme des Systems ein (vgl. Abschn. 2.4, 1. Absatz).
- d) FALSCH – Integration von neuen Funktionen während der Entwicklung ist kein „Wartungstest“. sondern erfordert Integrationstests (vgl. Abschn. 2.2.2).

12. Welche der folgenden Aussagen sind WAHR? [K2]

- A. Regressionstests und Abnahmetests sind gleich.**
- B. Regressionstests bestatigen, dass alle Fehlerzustande beseitigt sind.**
- C. Regressionstests sind gute Kandidaten fur die Testautomatisierung.**
- D. Regressionstests haben das Ziel nachzuweisen, dass durch anderungen am Code keine neuen Fehlerzustande eingebaut wurden.**
- E. Regressionstests sollten nicht als Teil des Integrationstests durchgefuhrt werden.**

a)	A und B sind wahr;	<input type="checkbox"/>
b)	A, C und E sind wahr;	<input type="checkbox"/>
c)	C und D sind wahr;	<input type="checkbox"/>
d)	B, D und E sind wahr;	<input type="checkbox"/>

Begrundung

Aussage A ist FALSCH – Ein Regressionstest ist der wiederholte Test eines bereits getesteten Programms nach einer anderung. Ziel ist es nachzuweisen, dass durch die anderungen keine Fehlerzustande eingebaut wurden (Abschn. 2.3.4, Absatz 1).

Aussage B ist FALSCH: Sie beschreibt einen „Fehlernachtest“ (Abschn. 2.3.4, 1. Absatz).

Aussage C ist WAHR: Regressionstestsuiten werden oft wiederholt und andern sich eher selten. Daher sind Regressionstests bevorzugte Kandidaten fur die Automatisierung (Abschn. 2.3.4, letzter Absatz).

Aussage D ist WAHR: Dies ist die Definition von Regressionstests (Abschn. 2.3.4, 2. Absatz, Satz 2).

Aussage E ist FALSCH: “Regressionstests konnen in allen Teststufen durchgefuhrt werden ...“ (Abschn. 2.3.4, letzter Absatz)

Also gilt:

- a) FALSCH, da A und B falsch sind
- b) FALSCH, da A und E falsch sind;
- c) KORREKT, da C und D wahr sind
- d) FALSCH, da B und E falsch sind

13. Welcher der folgenden Aussagen zum Vergleich zwischen Komponententest und Systemtest ist WAHR? [K2]

a)	Komponententests überprüfen die Funktion von Komponenten, Programmobjekten und Klassen, die separat prüfbar sind, während Systemtests die Schnittstellen zwischen den Komponenten und Wechselwirkungen mit anderen Teilen des Systems überprüfen.	<input type="checkbox"/>
b)	Testfälle für den Komponententest werden z. B. von den Komponenten-Spezifikationen abgeleitet, während Testfälle für den Systemtest z. B. von Anforderungsspezifikationen, funktionalen Spezifikationen und Anwendungsfällen abgeleitet werden.	<input type="checkbox"/>
c)	Komponententests konzentrieren sich nur auf die funktionalen Eigenschaften, während Systemtests sich auf die funktionalen und nicht-funktionalen Eigenschaften konzentrieren.	<input type="checkbox"/>
d)	Komponententests sind in der Verantwortung der Tester, während die Systemtests in der Regel in der Verantwortung der Benutzer des Systems liegen.	<input type="checkbox"/>

Begründung:

- a) FALSCH: Systemtests testen *nicht* die Schnittstellen und Wechselwirkungen zwischen den Komponenten und anderen Teilen des Systems; das ist Ziel des Integrationstests (vgl. Abschn. 2.2.2).
- b) KORREKT: Abschn. 2.2.1 (Komponententest), Testbasis, 1. dot; Abschn. 2.2.3 (Systemtest), Testbasis, 1. bis 3. dot.
- c) FALSCH: Komponententests konzentrieren sich nicht nur auf funktionale, sondern auch auf nicht-funktionale Aspekte (vgl. Abschn. 2.2.1, 2. Absatz nach „Typische Testobjekte“).
- d) FALSCH: Komponententests werden auch von Entwicklern durchgeführt (Abschn. 2.2.1, 3. Absatz nach „Typische Testobjekte“, Satz 2), wohingegen sich Tester mit Systemtests befassen (vgl. Abschn. 2.2.3, 4. Absatz nach „Typische Testobjekte“, Satz 2).

Fragen zum Thema
"Statischer Test"

14. Welches sind die Hauptaktivitäten eines formalen Reviews? [K1]

a)	Initialisierung, Status, Individuelle Vorbereitung, Reviewsitzung, Überarbeitung, Nachbereitung.	<input type="checkbox"/>
b)	Planung, individuelle Vorbereitung, Reviewsitzung, Überarbeitung, Abschluss, Nachbereitung.	<input type="checkbox"/>
c)	Planung, Kick-off, Individuelle Vorbereitung, Reviewsitzung, Überarbeitung, Nachbereitung.	<input type="checkbox"/>
d)	Individuelle Vorbereitung, Reviewsitzung, Überarbeitung, Abschluss, Nachbereitung, Grundursachenanalyse.	<input type="checkbox"/>

Begründung:

- a) FALSCH: Planung und Kick-off fehlen (vgl. Abschn. 3.2.1), Initialisierung und Status (vgl. etwa Abschn. 1.4.1, Testplanung und Steuerung) gehören nicht dazu;
- b) FALSCH: Kick-off fehlt (vgl. Abschn. 3.2.1);
- c) KORREKT (vgl. Abschn. 3.2.1);
- d) FALSCH: Planung fehlt (vgl. Abschn. 3.2.1); Grundursachenanalyse (vgl. Advanced Level Syllabus Testmanager, Abschn. 2.3.1.4 „Risikomanagement im Softwarelebenszyklus“, Absatz 2, Satz 4) gehört z.B. nicht dazu.

15. Welche der unten aufgeführten Reviewarten ist die GEEIGNETSTE zur Überprüfung sicherheitsrelevanter Komponenten in einem Software-Projekt, wenn zusätzlich auch das Review als formaler Prozess basierend auf Regeln und Checklisten nachgewiesen werden muss? [K2]

a)	Informelles Review	<input type="checkbox"/>
b)	Technisches Review	<input type="checkbox"/>
c)	Inspektion	<input type="checkbox"/>
d)	Walkthrough	<input type="checkbox"/>

Begründung

Für die Überprüfung der sicherheitsrelevanten Komponenten in einem Software-Projekt ist ein stärker geregeltes und dokumentiertes Review als formaler Prozess basierend auf Regeln und Checklisten erforderlich, also eine Inspektion.

Damit gilt:

- a) FALSCH
- b) FALSCH
- c) KORREKT
- d) FALSCH

16. Welche der folgenden Aussagen über die werkzeuggestützte statische Analyse ist FALSCH? [K1]

a)	Die werkzeuggestützte statische Analyse kann als vorbeugende Maßnahme genutzt werden, wenn sie durch einen entsprechend angemessenen Prozess unterstützt wird.	<input type="checkbox"/>
b)	Mit der werkzeuggestützten statischen Analyse können Fehlerzustände aufgedeckt werden, die nicht so leicht durch dynamische Tests gefunden werden.	<input type="checkbox"/>
c)	Die werkzeuggestützte statische Analyse kann durch frühe Fehlererkennung helfen, Kosten einzusparen.	<input type="checkbox"/>
d)	Die werkzeuggestützte statische Analyse ist gut dazu geeignet, um Fehlerwirkungen in der Software zu erzwingen.	<input type="checkbox"/>

Begründung:

a): FALSCH: Dieser Satz ist wahr, denn die werkzeuggestützte statische Analyse kann als vorbeugende Maßnahme verwendet werden (Abschn. 3.3, Vorteile, 2. und 6. dot).

b): FALSCH: Dieser Satz ist wahr, denn werkzeuggestützte statische Analyse kann Fehlerzustände finden, die durch dynamische Tests schwer zu finden sind (Abschn. 3.3, Hintergrund, 1. Absatz, Satz 3).

c): FALSCH: Dieser Satz ist wahr, denn werkzeuggestützte statische Analyse ist eine Methode, um Fehler(zustände) früh zu finden (Abschn. 3.3, Vorteile, 1. dot), was Kosten einspart.

d): KORREKT: Dieser Satz ist falsch, denn während der statische Analyse können keine Fehlerwirkungen entstehen, da der Code nicht ausgeführt wird (Abschn. 3.3, Hintergrund, Satz 2).

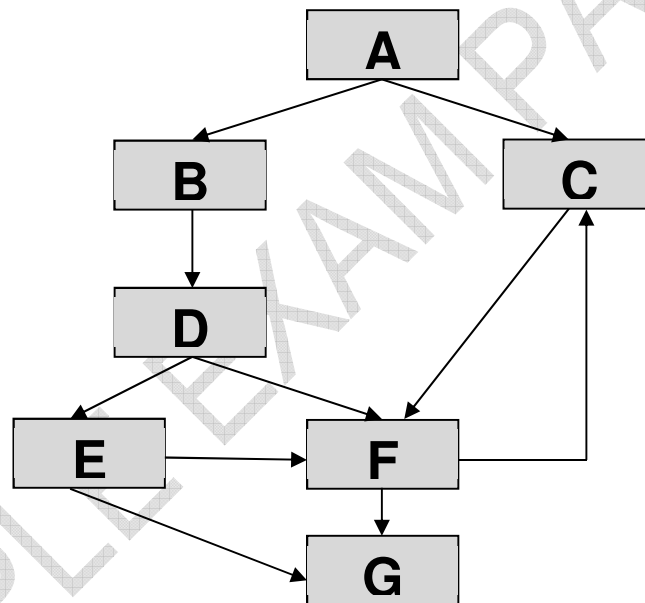
**Fragen zum Thema
"Testentwurfsverfahren"**

17. Eines der Testziele im Projekt ist es, eine Entscheidungsuberdeckung von 100% zu haben. Die folgenden drei Tests wurden fur die Ablaufsteuerung des unten abgebildeten Graphen ausgefuhrt. [K4]

Test_1 uberdeckt Pfad: A, B, D, E, G.

Test_2 uberdeckt Pfad: A, B, D, E, F, G.

Test_3 uberdeckt Pfad: A, C, F, C, F, C, F, G.



Welche der folgenden Aussagen ist in Bezug auf das Testziel Entscheidungsuberdeckung WAHR?

a)	Die Entscheidung D wurde nicht vollstandig getestet.	<input type="checkbox"/>
b)	100% Entscheidungsuberdeckung ist erreicht worden.	<input type="checkbox"/>
c)	Die Entscheidung E wurde nicht vollstandig getestet.	<input type="checkbox"/>
d)	Die Entscheidung F wurde nicht vollstandig getestet.	<input type="checkbox"/>

Begründung

In dem Diagramm gibt es die folgenden vier Bedingungen/Entscheidungen: A, D, E, F.
Der Test_1 deckt A-> B, D-> E und E-> G ab.
Der Test_2 deckt A-> B, D-> E, E-> F und F-> G ab.
Der Test_3 deckt A-> C, F-> C und F-> G ab.
Daher wird Entscheidung A überdeckt (A-> B durch Test_1 und A-> C durch Test_3),
Entscheidung E ist überdeckt (E-> G durch Test_1 und E-> F durch Test_2), Entscheidung
F ist überdeckt (F-> C durch Test_3 und F-> G durch Test_2 und Test_3).
Entscheidung D ist nicht überdeckt: zwar D-> E durch Test_1 und Test_2, aber D-> F ist
nicht überdeckt.

Daher gilt:

- a) WAHR: D wurde nicht überdeckt (siehe oben);
- b) FALSCH: 1 von 4 Entscheidungen (D) wurde nicht überdeckt (s. oben), also gibt es nur 75% Entscheidungsüberdeckung;
- c) FALSCH: E wurde vollständig getestet/überdeckt (s. oben);
- d) FALSCH: F wurde vollständig getestet/überdeckt (s. oben).

18. Die folgende Fehlerwirkung wurde während des Testens gefunden:
Beim Empfangen von Kundendaten durch den Server stürzte das System ab. Der Fehlerzustand wurde durch eine Korrektur im Code, der die Verfügbarkeit des Netzes während der Datenübertragung überprüft, behoben. Die bestehenden Testfälle decken 100% aller Anweisungen des entsprechenden Moduls ab. Um die Korrektur zu überprüfen und eine höhere Testabdeckung zu erreichen, wurden einige neue Tests entwickelt und zur bestehenden Testsuite hinzugefügt und ausgeführt. [K2]

Welche Arten von Tests sind oben erwähnt?

- A. Funktionale Tests
- B. Strukturelle Tests
- C. Fehlernachtest
- D. Performanztest

a)	A und B sind erwähnt – C und D nicht.	<input type="checkbox"/>
b)	A und C sind erwähnt – B und D nicht.	<input type="checkbox"/>
c)	A, B und C sind erwähnt – D nicht.	<input type="checkbox"/>
d)	B, C und D sind erwähnt – A nicht.	<input type="checkbox"/>

Begründung:

A) WAHR: Empfangen von Kundendaten (Satz 2) ist ein typischer funktionaler Test.

B) WAHR: In der Problembeschreibung (Satz 4) heißt es: "Die bestehenden Testfälle decken 100% aller Anweisungen des entsprechenden Moduls ab"; Anweisungstest ist struktureller Test (vgl. Abschn. 4.4.1).

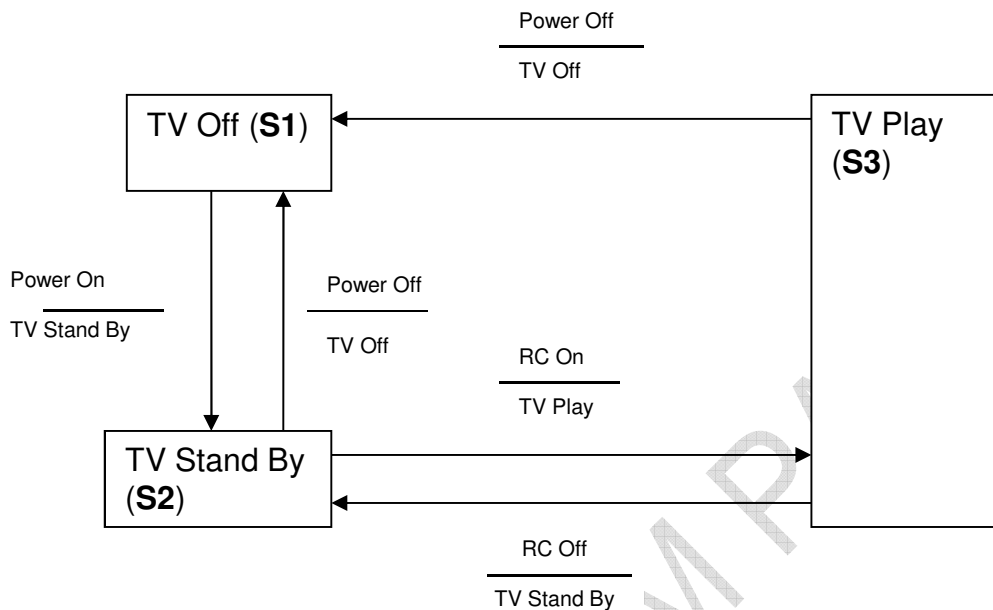
C) WAHR: In der Problembeschreibung (Satz 5) heißt es: "Um die Korrektur zu überprüfen und eine höhere Testabdeckung zu erreichen, wurden einige neue Tests entwickelt und ... ausgeführt"; das ist ein Fehlernachttest (vgl. Abschn. 2.3.4, 1. Absatz).

D) FALSCH: In der Beschreibung des Problems gibt es keine Informationen über eine Leistungsüberprüfung.

Daher gilt:

- a) FALSCH, da C fehlt;
- b) FALSCH, da B fehlt;
- c) KORREKT, da A, B und C erwähnt sind;
- d) FALSCH, da A fehlt und D fälschlich erwähnt ist.

19. Welche der folgenden Aussagen zum Zustandsübergangsdiagramm und der dargestellten Tabelle von Testfällen ist WAHR? [K3]



Testfall	1	2	3	4	5
Startzustand	S1	S2	S2	S3	S3
Eingabe	Power On	Power Off	RC On	RC Off	Power Off
Erwartete Ausgabe	TV Stand By	TV Off	TV Play	TV Stand By	TV Off
Endzustand	S2	S1	S3	S2	S1

a)	Die Testfälle decken sowohl gültige und ungültige Übergänge des Zustandsübergangsdiagramms ab.	<input type="checkbox"/>
b)	Die Testfälle decken alle gültigen Übergänge des Zustandsübergangsdiagramms ab.	<input type="checkbox"/>
c)	Die Testfälle decken nur einige der möglichen Übergänge im Zustandsübergangsdiagramm ab.	<input type="checkbox"/>
d)	Die Testfälle decken Paare von Übergängen im Zustandsübergangsdiagramm ab.	<input type="checkbox"/>

Begründung

Die vorgeschlagenen Testfälle überdecken genau alle fünf möglichen gültigen Übergänge im gegebenen Zustandsdiagramm (S1-> S2, S2-> S1, S2-> S3, S3-> S2, S3-> S1).

Daher gilt:

- a) FALSCH, da keine ungültigen Übergänge abgedeckt werden.
- b) KORREKT, da alle gültigen Übergänge abgedeckt werden.
- c) FALSCH, da alle gültigen Übergänge abgedeckt werden.
- d) FALSCH, da die Testfälle keine Paare von Übergängen enthalten.

20. Welche der folgenden Aussagen zur Äquivalenzklassenbildung sind WAHR? [K2]

- A. Äquivalenzklassenbildung teilt mögliche Eingabewerte in Gruppen, von deren Elementen jeweils gleiches Verhalten erwartet wird.
- B. Äquivalenzklassenbildung verwendet sowohl Äquivalenzklassen mit gültigen als auch solche mit ungültigen Daten.
- C. Äquivalenzklassenbildung muss mindestens zwei Werte aus jeder Äquivalenzklasse enthalten.
- D. Äquivalenzklassenbildung kann nur für die Prüfung der Eingaben einer graphischen Benutzeroberfläche verwendet werden.

a)	A, B und D sind wahr; C ist falsch.	<input type="checkbox"/>
b)	A ist wahr; B, C und D sind falsch.	<input type="checkbox"/>
c)	B und C sind wahr; A und D sind falsch.	<input type="checkbox"/>
d)	A und B sind wahr; C und D sind falsch.	<input type="checkbox"/>

Begründung

A ist wahr, denn Äquivalenzklassenbildung teilt mögliche Eingaben in Gruppen ein, bei denen erwartet wird, dass alle Elemente das gleiche Verhalten bewirken (Abschn. 4.3.1, 1. Satz).

B ist wahr, denn Äquivalenzklassenbildung verwendet/bildet Klassen mit gültigen als auch mit ungültigen Daten (Abschn. 4.3.1, 2. Satz).

C ist falsch, denn Äquivalenzklassen müssen nur abgedeckt werden (Abschn. 4.3.1, 1. Absatz, 2. letzter Satz); dafür ist es genug, einen Wert aus jeder Klasse zu verwenden.

D ist falsch: Eingaben über eine GUI sind möglich, aber das ist nicht nötig.

Daher gilt:

- a) FALSCH, da D nicht wahr ist;
- b) FALSCH, da B nicht falsch ist;
- c) FALSCH, da C nicht wahr und A nicht falsch ist;
- d) KORREKT, da A und B wahr und C und D falsch sind.

21. Welche der folgenden Antworten enthält nur Verfahren, die als Black-Box-Entwurfsverfahren kategorisiert werden können? [K1]

a)	Äquivalenzklassenbildung, Entscheidungstabellentest, Zustandsbasierter Test und Grenzwertanalyse.	<input type="checkbox"/>
b)	Äquivalenzklassenbildung, Entscheidungstabellentest, Anweisungsüberdeckungstest, Anwendungsfallbasierter Test.	<input type="checkbox"/>
c)	Äquivalenzklassenbildung, Entscheidungsüberdeckungstest, Anwendungsfallbasierter Test.	<input type="checkbox"/>
d)	Äquivalenzklassenbildung, Entscheidungsüberdeckungstest, Grenzwertanalyse.	<input type="checkbox"/>

Begründung

- a) KORREKT, denn alle vier Verfahren sind Black-Box-Verfahren (vgl. Abschnitt 4.3).
- b) FALSCH, denn Anweisungsüberdeckung ist ein White-Box-Verfahren (Abschn. 4.4.1).
- c) FALSCH, denn Entscheidungsüberdeckung ist ein White-Box-Verfahren (Abschn. 4.4.2).
- d) FALSCH, denn Entscheidungsüberdeckung ist ein White-Box-Verfahren (Abschn. 4.4.2).

22. Der Bonus eines Mitarbeiters soll berechnet werden. Der Bonus kann nicht negativ, aber Null sein. Der Bonus hängt von der Anstellungsdauer ab:

Ein Mitarbeiter kann weniger als oder gleich 2 Jahre, mehr als 2 Jahre aber weniger als 5 Jahre, 5 bis inklusive 10 Jahre oder länger als 10 Jahre angestellt sein. Abhängig von der Dauer der Anstellung wird ein Mitarbeiter entweder keinen Bonus oder einen Bonus von 10%, 25% bzw. 35% erhalten. [K3]

Wie viele gültige Äquivalenzklassen sind notwendig, um die Berechnung des Bonus zu testen?

a)	3	<input type="checkbox"/>
b)	5	<input type="checkbox"/>
c)	2	<input type="checkbox"/>
d)	4	<input type="checkbox"/>

Begründung

- a) FALSCH, eine zu wenig (siehe die 4 KORREKTEN Äquivalenzklassen in d)
b) FALSCH, eine zu viel (siehe die 4 KORREKTEN Äquivalenzklassen in d)
c) FALSCH, zwei zu wenig (siehe die 4 KORREKTEN Äquivalenzklassen in d)
d) KORREKT. Die 4 Äquivalenzklassen entsprechen der Beschreibung in der Frage:

1. Klasse: $0 \leq \text{Beschäftigungszeit} \leq 2$,
2. Klasse: $2 < \text{Beschäftigungszeit} < 5$,
3. Klasse: $5 \leq \text{Beschäftigungszeit} \leq 10$,
4. Klasse: $10 < \text{Beschäftigungszeit}$

- 23. Welche der folgenden Aussagen über die Vorteile der Ableitung von Testfällen aus Anwendungsfällen sind wahr und welche sind falsch? [K2]**

Das Ableiten von Testfällen aus Anwendungsfällen ...

- A. ... ist hilfreich für System- und Abnahmetests.**
- B. ... ist nur für automatisierte Tests hilfreich.**
- C. ... ist hilfreich für Komponententests.**
- D. ... ist hilfreich für die Prüfung von Integrationstests.**

a)	A und D sind wahr; B und C sind falsch.	<input type="checkbox"/>
b)	A ist wahr; B, C und D sind falsch.	<input type="checkbox"/>
c)	B und D sind wahr; A und C sind falsch.	<input type="checkbox"/>
d)	A, C und D sind wahr; B ist falsch.	<input type="checkbox"/>

Begründung

Satz A ist WAHR: Anwendungsfälle beschreiben, wie Benutzer mit dem fertigen System interagieren, passen daher am besten für die Definition von Tests auf der Systemebene (Abschn. 4.3.5, 1. Absatz). Darüber hinaus gilt: „Anwendungsfälle sind für den Entwurf von Abnahmetests mit Kunden-/Anwenderbeteiligung sehr hilfreich“ (Abschn. 4.3.5, 2. Absatz, Satz 3). "

Satz B ist FALSCH: Anwendungsfälle können manuell ausgeführt werden, nicht nur automatisch. (Jedenfalls wird das im Lehrplan, Abschn. 4.3.5, nicht behauptet.)

Satz C ist FALSCH: Auf der Komponenten-Ebene werden Testfälle „von Entwicklungsdokumenten wie einer Komponententestspezifikation, dem Softwareentwurf oder dem Datenmodell abgeleitet.“ (Abschn. 2.2.1, 2. Absatz nach „Typische Testobjekte“, Satz 2).

Satz D ist WAHR, denn die Anwendungsfälle „können ... auch Fehlerzustände im Umfeld der Integration aufdecken, die durch den Test der einzelnen Komponenten nicht gefunden werden könnten“ (Abschn. 4.3.5, 2. Absatz, Satz 4).

Daher gilt:

- a) KORREKT, da genau A und D wahr sind;
- b) FALSCH, da D wahr (und nicht falsch) ist;
- c) FALSCH, da B falsch (und nicht wahr) ist und A wahr (und nicht falsch) ist;
- d) FALSCH, da C falsch (und nicht wahr) ist.

24. Was ist die empfohlene Grundlage für das Testen mit Fehlerangriffen?
[K2]

a)	Erfahrung, gesammelte Daten und Wissen über Fehlerwirkungen und -zustände	<input type="checkbox"/>
b)	Eine zu Beginn des Projekts durchgeführte Risikoidentifikation	<input type="checkbox"/>
c)	Von Fachexperten abgeleitete, anwendungsfallbasierte Tests	<input type="checkbox"/>
d)	Erwartete Ergebnisse aus dem Vergleich mit einem bestehenden System	<input type="checkbox"/>

Begründung

Begründung:

a) KORREKT: "Die Liste der Fehlerzustände und Fehlerwirkungen kann erstellt werden auf der Basis von Erfahrungen, verfügbaren Daten über Fehlerzustände und Fehlerwirkungen ...“ (Abschn. 4.5, 2. Absatz, letzter Satz).

b) FALSCH : Risikoidentifikation wählt die betreffenden Bereiche des Projekts aus - nicht, wie sie zu testen sind.

c) FALSCH: Das Testen auf der Basis von Geschäftsprozessen zielt nicht auf mögliche Schwächen in der Software, die aus Erfahrung und Allgemeinwissen bekannt sind, was beim Test mit Fehlerangriffen der Ansatz ist (Abschn. 4.5, 2. Absatz, letzter Satz). Es versucht nur zu überprüfen, ob bestimmte Anwendungsfälle ausgeführt werden können.

d) FALSCH: Der Satz beschreibt eine der Möglichkeiten, Testfälle zu ermitteln. Der Ansatz hat aber nichts spezifisches mit dem Ansatz „Fehlerangriffe“ zu tun, bei dem eine Liste der möglichen Fehlerzustände und Fehlerwirkungen auf der Basis von Erfahrungen erstellt wird (Abschn. 4.5, 2. Absatz, letzter Satz).

25. Welches der folgenden Testverfahren ist AM BESTEN geeignet, wenn unzureichende Spezifikationen und Zeitdruck gegeben sind? [K1]

a)	Anwendungsfallbasierter Test	<input type="checkbox"/>
b)	Anweisungstest	<input type="checkbox"/>
c)	Exploratives Testen	<input type="checkbox"/>
d)	Entscheidungstest	<input type="checkbox"/>

Begründung

a) FALSCH, das Projekt hat eine schlechte Spezifikationen, daher gibt es nur eine geringe Möglichkeit, dass irgendwelche Anwendungsfälle bestehen.

b) FALSCH: Anweisungstest ist zeitaufwendig, und es gibt Zeitdruck im Projekt.

c) KORREKT: Das explorative Testen ist angebracht, wenn es schlechte Dokumentation und Zeitdruck gibt (Abschn. 4.5, Hintergrund, 3. Absatz, Satz 3).

d) FALSCH: Entscheidungstest ist zeitaufwendig, und es gibt Zeitdruck im Projekt.

26. Welches der folgenden Testverfahren ist ein White-Box-Testverfahren? [K1]

a)	Entscheidungstests	<input type="checkbox"/>
b)	Grenzwertanalyse	<input type="checkbox"/>
c)	Äquivalenzklassenbildung	<input type="checkbox"/>
d)	Zustandsbasierter Test	<input type="checkbox"/>

Begründung

- a) KORREKT: Entscheidungstests sind ein White-Box-Verfahren (Abschn. 4.4.2 in Kap. 4.4).
- b) FALSCH: Grenzwertanalyse ist ein Black-Box-Verfahren (Abschn. 4.3.2 in Kap. 4.3).
- c) FALSCH: Äquivalenzklassenbildung ist ein Black-Box-Verfahren (Abschn. 4.3.1).
- d) FALSCH: Zustandsbasierter Test ist ein Black-Box-Verfahren (Abschn. 4.3.4 in Kap. 4.3).

27. Sie haben mit dem spezifikationsorientierten Testen eines Programms begonnen. Das Programm berechnet den größten gemeinsamen Teiler (GGT) zweier Integer-Zahlen (A und B) größer als Null. [K3]

calcGGT (A, B);

Die folgenden Testeingaben wurden spezifiziert:

Testfall	A	B
1	1	1
2	INT_MAX	INT_MAX
3	1	0
4	0	1
5	INT_MAX + 1	1
6	1	INT_MAX + 1

INT_MAX: größte Integer-Zahl

Welches Testverfahren wurde angewandt, um die Testfälle 1 bis 6 zu bestimmen?

a)	Grenzwertanalyse	<input type="checkbox"/>
b)	Zustandsbasierter Test	<input type="checkbox"/>
c)	Anwendungsfallbasierter Test	<input type="checkbox"/>
d)	Entscheidungstabellentest	<input type="checkbox"/>

Begründung

- a) Korrekt: Die angegebenen Werte sind Grenzwerte an der unteren Grenze (1) und an der oberen Grenze (INT_MAX) des gültigen Bereichs 1 bis INT_MAX bzw. Grenzwerte der ungültigen Bereiche „<1“ (0) bzw. „>INT_MAX“ (INT_MAX+1) (s. Abschn. 4.3.2, 1. Absatz).
- b) FALSCH: Die angegebenen Werte sind keine Zustandswerte, es gibt keine Zustandsübergänge (Abschn. 4.3.4, 1. Absatz).
- c) FALSCH: Die angegebenen Werte sind keine Anwendungsfälle mit Vor- und Nachbedingungen. (Abschn. 4.3.5, 1. Absatz).
- d) FALSCH: Die angegebenen Werte sind keine logischen Bedingungen. (Abschn. 4.3.3, 1. Satz).

28. Den Mitarbeitern einer Firma werden Prämien ausbezahlt, wenn sie länger als ein Jahr im Unternehmen beschäftigt sind und individuell vereinbarte Ziele erreichen.

Dieser Sachverhalt lässt sich in einer Entscheidungstabelle darstellen:

		T1	T2	T3	T4	T5	T6	T7	T8
Bedingungen									
Beding1	Beschäftigung länger als 1 Jahr?	JA	NEIN	JA	NEIN	JA	NEIN	JA	NEIN
Beding2	Ziel abgestimmt?	NEIN	NEIN	JA	JA	NEIN	NEIN	JA	JA
Beding3	Ziel erreicht?	NEIN	NEIN	NEIN	NEIN	JA	JA	JA	JA
Aktion									
	Auszahlung der Jahresprämie?	NEIN	NEIN	NEIN	NEIN	NEIN	NEIN	JA	NEIN

Welche Testfälle können in der oben aufgeführten Entscheidungstabelle entfernt werden, weil diese Situation in der Praxis nicht eintreten kann und sie damit redundant sind? [K3]

a)	T1 and T2	<input type="checkbox"/>
b)	T3 and T4	<input type="checkbox"/>
c)	T7 and T8	<input type="checkbox"/>
d)	T5 and T6	<input type="checkbox"/>

Begründung

a) FALSCH: Es ist mit T1 und T2 zu prüfen, ob die Prämie nicht gezahlt wird, wenn keine Ziele vereinbart/abgestimmt sind (und natürlich nicht erreicht wurden).

b) FALSCH: Es ist mit T3 und T4 zu prüfen, ob die Prämie nicht gezahlt wird, wenn vereinbarte/abgestimmte Ziele nicht erreicht wurden.

c) FALSCH: Es ist mit T7 und T8 zu prüfen, ob die Prämie gezahlt oder nicht gezahlt wird, wenn Ziele vereinbart und erreicht wurden und die Beschäftigungszeit länger als 1 Jahr (T7) (oder nicht, T8) ist.

d) KORREKT: Die Testfälle T5 und T6 beschreiben die Situation, wobei das Ziel erreicht ist - allerdings nicht vereinbart/abgestimmt wurde. Diese Situation kann nicht auftreten, deshalb können wir die entsprechenden Tests T5 und T6 weglassen.

Fragen zum Thema
"Testmanagement"

29. Welche der folgenden Aussagen beschreibt AM BESTEN die Aufgabenteilung zwischen Testmanager und Tester? [K1]

a)	Der Testmanager plant Testaktivitäten und wählt die Normen aus, die befolgt werden müssen, während der Tester die Werkzeuge und Steuerelemente auswählt, die verwendet werden.	<input type="checkbox"/>
b)	Der Testmanager plant, organisiert und steuert die Testaktivitäten, während der Tester Testfälle spezifiziert und ausführt.	<input type="checkbox"/>
c)	Der Testmanager plant, überwacht und steuert die Testaktivitäten, während der Tester die Tests entwirft und über die Freigabe des Testobjekts entscheidet.	<input type="checkbox"/>
d)	Der Testmanager plant und organisiert die Testdurchführung und spezifiziert die Testfälle, während die Tester die Tests priorisieren und durchführen.	<input type="checkbox"/>

Begründung

- a) FALSCH: Auswahl der Werkzeuge ist eine Testmanager-Aufgabe (Abschn. 5.1.2, 10./ drittletzter dot).
- b) KORREKT - siehe Abschnitt 5.1.2 (Testmanager, 4. dot, Tester, 3.+5. dot).
- c) FALSCH: Die Entscheidung über die Freigabe des Testobjekts ist ein Aufgabe des Testmanagers (Abschn. 5.1.2, 5. dot).
- d) FALSCH: Der Tester muss die Testfälle spezifizieren (Abschn. 5.1.2, 3. dot).

30. Welche der folgenden Risiken können als Produktrisiken kategorisiert werden? [K1]

a)	Geringe Qualität der Anforderungen, des Designs, des Codes und der Testfälle.	<input type="checkbox"/>
b)	Politische Probleme und Verzögerungen bei besonders komplexen Teilen des Produkts.	<input type="checkbox"/>
c)	Fehleranfällige Komponenten, möglicher Schaden für den Menschen, schlechte Softwareeigenschaften.	<input type="checkbox"/>
d)	Probleme bei der Definition korrekter Anforderungen, fehleranfällige Bereiche der Software oder des Systems.	<input type="checkbox"/>

Begründung

- a) FALSCH: Geringe Qualitätsanforderungen sind ein Projektrisiko (Abschnitt 5.5.1, technische Aspekte, 5. dot)
- b) FALSCH: Alle Angaben in dieser Option sind Projektrisiken (Abschnitt 5.5.1, organisatorische Faktoren, 3. dot; technische Aspekte, 3.+4. dot)
- c) KORREKT: Alle Angaben sind Produktrisiken (Abschnitt 5.5.2, 1. bis 3. dot)
- d) FALSCH: Anforderungs-, Software- und System-Probleme sind Projektrisiken (Abschnitt 5.5.1, technische Aspekte, 1.+5. dot).

31. Welche der folgenden Listen enthält nur typische Endekriterien? [K2]

a)	Kennzahlen zu Zuverlassigkeit und Testuberdeckung, Testkosten, Zeitplan, Status uber Fehlerbehebung und Restrisiken	<input type="checkbox"/>
b)	Kennzahlen zu Zuverlassigkeit und Testuberdeckung, Grad der Unabhangigkeit der Tester und Grad der Produktvollstandigkeit	<input type="checkbox"/>
c)	Kennzahlen zu Zuverlassigkeit und Testuberdeckung, Testkosten, Verfugbarkeit von testbarem Code, Zeit bis Markteinfuhrung, Grad der Produktvollstandigkeit	<input type="checkbox"/>
d)	Zeit bis Markteinfuhrung, Restfehler, Qualifikation der Tester, Grad der Unabhangigkeit der Tester, Testuberdeckung und Testkosten	<input type="checkbox"/>

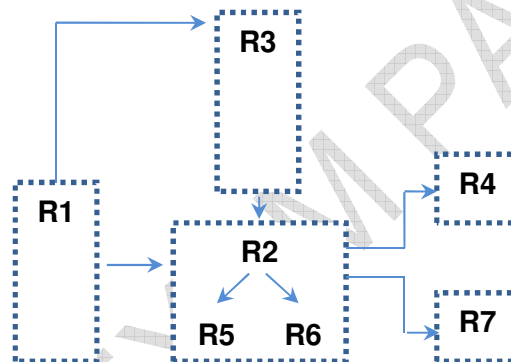
Begrundung

- a) KORREKT: siehe Abschnitt 5.2.4 (alle 5 Punkte)
- b) FALSCH: Der Grad der Unabhangigkeit der Tester spielt keine Rolle bei den Endekriterien (vgl. Abschn. 5.2.4).
- c) FALSCH : Die "Verfugbarkeit von testbarem Code" ist ein Testeingangskriterium (Abschn. 5.2.3, 3. dot)
- d) FALSCH: Der Grad der Unabhangigkeit der Tester sowie die Qualifikation der Tester spielt keine Rolle bei den Endekriterien (vgl. Abschn. 5.2.4).

32. Als Testmanager verantworten Sie den Test folgender Anforderungen:

- R1 - Prozessanomalien
- R2 - Synchronisation
- R3 - Zulassung
- R4 - Problembearbeitung
- R5 - Finanzdaten
- R6 - Diagramm Daten
- R7 - Änderungen am Benutzerprofil

Logische Anforderungsabhängigkeiten (A -> B bedeutet, dass B von A abhängig ist):



Wie würden Sie den Testausführungsplan, entsprechend der Abhängigkeiten der Anforderungen, aufbauen? [K3]

a)	R3 -> R2 -> R1 -> R7 -> R5 -> R6 -> R4	<input type="checkbox"/>
b)	R2 -> R5 -> R6 -> R4 -> R7 -> R1 -> R3	<input type="checkbox"/>
c)	R1 -> R3 -> R2 -> R5 -> R6 -> R4 -> R7.	<input type="checkbox"/>
d)	R1 -> R2 -> R5 -> R6 -> R3 -> R4-> R7.	<input type="checkbox"/>

Begründung

- a) FALSCH: Alles hängt von R1 ab, also ist jeder Testlauf, der nicht mit R1 beginnt, falsch.
- b) FALSCH: Begründung wie bei a).
- c) KORREKT: Die Tests sind in einer Reihenfolge festgelegt, welche alle Abhängigkeiten berücksichtigt.
- d) FALSCH: R2 ist abhängig von R3, also sollte R3 vor R2 getestet werden.

33. Identifizieren Sie einen m3glichen Vorteil des unabh3ngigen Testens?
[K1]

a)	Die Arbeit ist effizienter, da die Tester die Entwickler nicht st3ndig st3ren.	<input type="checkbox"/>
b)	Unabh3ngige Tester sind tendenziell unvoreingenommen und finden andere Fehler als die Entwickler.	<input type="checkbox"/>
c)	Unabh3ngige Tester brauchen keine zus3tzliche Ausbildung oder Schulung.	<input type="checkbox"/>
d)	Unabh3ngige Tester reduzieren den Ressourcen-Engpass im Fehler- und Abweichungsmanagementprozess.	<input type="checkbox"/>

Begr3ndung

- a) FALSCH: Unabh3ngigkeit bedeutet nicht „keine Zusammenarbeit“.
- b) KORREKT: Dies ist ein Grund f3r die Unabh3ngigkeit (Abschn. 5.1.1, erster Vorteil).
- c) FALSCH: Nat3rlich ben3tigen Tester Ausbildung und Schulung.
- d) FALSCH: Es gibt keine Verbindung zwischen der Unabh3ngigkeit der Tester und dem Engpass im Fehlermanagement- und Abweichungsmanagement-Prozess.

34. Welches der folgenden Elemente wird als ein Projektrisiko eingestuft? [K1]

a)	Mangel an sachkundigem Personal	<input type="checkbox"/>
b)	Schlechte Softwareeigenschaften (z. B. Benutzbarkeit)	<input type="checkbox"/>
c)	Fehleranfällige Software wurde geliefert.	<input type="checkbox"/>
d)	Mangelhafte Zuverlässigkeit	<input type="checkbox"/>

Begründung

- a) KORREKT: Knappheit von Sachkunde und Personal ist ein typisches Projektrisiko (Abschn. 5.5.1, Organisatorische Faktoren, 1. dot)
- b) FALSCH: Schlechte SW-Eigenschaften sind ein Produktrisiko (Abschn. 5.5.2, 3. dot)
- c) FALSCH: Lieferung fehleranfälliger SW ist ein Produktrisiko (Abschn. 5.5.2, 1. dot)
- d) FALSCH: Mangelhafte Zuverlässigkeit ist ein Produktrisiko (Abschn. 5.5.2, 3. dot).

35. Als Testmanager sind Sie aufgefordert, einen Testabschlussbericht abzuliefern.

Was sollten Sie gemäß IEEE-829 bezüglich Testaktivitäten in Ihrem Bericht berücksichtigen? [K2]

a)	Die Anzahl der ausgeführten Testfälle und Ihre Ergebnisse.	<input type="checkbox"/>
b)	Eine Zusammenfassung der wichtigsten Testaktivitäten, Ereignisse und deren Status in Bezug auf die vereinbarten Ziele.	<input type="checkbox"/>
c)	Eine Gesamtbewertung aller Entwicklungsartefakte.	<input type="checkbox"/>
d)	Das Training der Mitglieder des Testteams, um die Effektivität des Testens zu erhöhen.	<input type="checkbox"/>

Begründung

- a) FALSCH: Die Anzahl der durchgeführten Testfälle und ihre Ergebnisse können in einem Testabschlussbericht nach IEEE 829 enthalten sein, aber sind kein wichtiger Teil.
- b) KORREKT: Der Testabschlussbericht muss Informationen zu den wichtigsten Testaktivitäten, Ereignissen und ihren Status in Bezug auf die vereinbarten Ziele enthalten (vgl. auch Lehrplan, Abschn. 5.3.2, 1. dot und dort zu Metriken, 3. dot).
- c) FALSCH: Die Bewertung der einzelnen Entwicklungsarbeiten ist NICHT Teil des Testabschlussberichts nach IEEE 829.
- d) FALSCH: Zusätzliche Ausbildung ist nicht die Information, die in dem Testabschlussbericht nach IEEE 829 enthalten sein sollte.

- 36. Sie sind Tester in einem sicherheitskritischen Software-Entwicklungs-Projekt. Bei der Ausföhrung eines Tests trifft eines der erwarteten Ergebnisse nicht zu. Sie erstellen dazu einen Fehler- und Abweichungsbericht.**

Welche Informationen sind gemäÙ IEEE-Standard 829 für die Fehlererfassung am wichtigsten? [K3]

a)	Beschreibung der Auswirkung, Problembeschreibung, Datum, Ihr Name	<input type="checkbox"/>
b)	Eindeutige ID des Abweichungsberichtes, speziell benötigte Ausgangslage, Fehlerverursacher	<input type="checkbox"/>
c)	Übertragene Gegenstände, Ihren Namen und Ihre Vermutung der möglichen Fehlerursachen	<input type="checkbox"/>
d)	Problembeschreibung, Entwicklungsumgebung, erwartete Testergebnisse	<input type="checkbox"/>

Begründung

- a) **KORREKT:** Die wichtigste Information, die in den Fehler- und Abweichungsbericht für kritische Software aufgenommen werden sollte, ist die Auswirkung (auf das System und die Stakeholder-Interessen, Abschn. 5.6, Informationen, dot 6 und 7).
- b) **FALSCH:** Diese Informationen können im Fehler- und Abweichungsbericht stehen, aber sie sind nicht so wichtig wie die Auswirkungen (und daher auch nicht in Abschn. 5.6 enthalten).
- c) **FALSCH:** Der Fehler- und Abweichungsbericht sollte Sachinformationen enthalten, nicht des Testers Vermutungen über die möglichen Fehlerursachen (was daher auch nicht in Abschn. 5.6 enthalten ist).
- d) **FALSCH:** Diese Informationen können im Fehler- und Abweichungsbericht stehen (Abschn. 5.6, Informationen, dot 2 (erwartete Testergebnisse), dot 5 (Beschreibung der Abweichung)), aber sie sind nicht so wichtig wie die Auswirkungen (und die Entwicklungsumgebung ist daher auch nicht in Abschn. 5.6 enthalten).

Fragen zum Thema
"Testwerkzeuge"

37. Wählen Sie aus der folgenden Liste die empfohlenen Vorgehensweisen aus, um ein ausgewähltes Testwerkzeug in eine Organisation einzuführen: [K1]

1. Das Testwerkzeug wird für die gesamte Organisation zur gleichen Zeit in Betrieb genommen.
2. Beginnen Sie mit einem Pilotprojekt.
3. Die Prozesse werden auf die Verwendung des Werkzeugs angepasst und verbessert.
4. Durchführung von Schulungen und Coaching für neue Benutzer.
5. Lassen Sie jedes Team selber entscheiden, wie sie das Tool am besten nutzen.
6. Überwachung der Kosten, damit diese nicht die anfänglichen Anschaffungskosten übersteigen.
7. Zusammentragen der Erfahrungen aus allen Teams.

a)	1,3,4,5	<input type="checkbox"/>
b)	2,5,6	<input type="checkbox"/>
c)	2,3,4,7	<input type="checkbox"/>
d)	1,6,7	<input type="checkbox"/>

Begründung

Begründung:

Satz 1 ist inkorrekt: Es wird empfohlen, zunächst ein Pilotprojekt durchzuführen, bevor das Testwerkzeug schrittweise in der gesamten Organisation in Betrieb genommen wird (Abschnitt 6.3, Erfolgsfaktoren, 1. dot).

Satz 2 ist KORREKT: Siehe Abschn. 6.3, 2. Absatz, Satz 1: „Die Einführung des ausgewählten Werkzeugs in einer Organisation beginnt mit einem Pilotprojekt“.

Satz 3 ist KORREKT: "Bewertung, wie das Werkzeug mit den existierenden Werkzeugen und Prozessen zusammenpasst" (Abschn. 6.3, Pilotprojekt-Ziele, 2. dot).

Satz 4 ist KORREKT: Bereitstellung von Trainingsmaßnahmen und Coaching ist einer (der 3.) der Erfolgsfaktoren der Inbetriebnahme (in Abschnitt 6.3 dort 3. dot).

Satz 5 ist inkorrekt: Wenn Sie jedes Team entscheiden lassen, wie es das Tool verwendet, wird es ein Durcheinander geben. "Richtlinien für die Werkzeugbenutzung definieren" ist einer der Erfolgsfaktoren für den Einsatz (Abschnitt 6.3, 4. Erfolgsfaktor).

Satz 6 ist inkorrekt: Die Kosten für den Einsatz eines Werkzeuges sind mehr als nur die Anschaffungskosten für das Werkzeug. Dies nicht zu realisieren ist eines der Risiken, die der Werkzeugbereitstellung zugeordnet sind (Abschnitt 6.2.1, Risiken, 2. dot).

Satz 7 ist KORREKT: "Es wird ein Erfahrungskatalog erstellt, basierend auf den Erfahrungen aller Teams." ist einer der Erfolgsfaktoren für den Einsatz (Abschnitt 6.3, 8. Erfolgsfaktor)

Daher gilt:

- a) (1,3,4,5): FALSCH wg. 1 und 5,
- b) (2,5,6): FALSCH wg. 5 und 6,
- c) (2,3,4,7): **KORREKT**,
- d) (1,6,7): FALSCH wg. 1 und 6.

38. Welche der folgenden Eigenschaften beschreiben AM BESTEN ein schlüsselwortgetriebenes Testausföhrungswerkzeug? [K1]

a)	Eine Tabelle mit Testeingangsdaten, Schlösselwörtern und den erwarteten Ergebnissen steuert die Ausführung des zu testenden Systems.	<input type="checkbox"/>
b)	Aktionen von Testern, die in einem Skript aufgezeichnet wurden und mehrfach ausgeföhrte werden.	<input type="checkbox"/>
c)	Aktionen von Testern, die in einem Skript aufgezeichnet wurden, das danach mit mehreren Sets von Test-Eingabedaten ausgeföhrte wird.	<input type="checkbox"/>
d)	Die Möglichkeit, Testergebnisse aufzuzeichnen und sie danach mit den erwarteten Ergebnissen zu vergleichen, welche in einer Textdatei gespeichert sind.	<input type="checkbox"/>

Begründung

- a) KORREKT - "In einem schlüsselwortgetriebenen Testansatz ... enthält ein Tabellenblatt zusätzlich zu den Testdaten Schlösselwörter (auch Aktionswörter genannt), welche die auszuföhrenden Aktionen beschreiben" (Abschnitt 6.2.2, 5. Absatz).
- b) FALSCH: Das ist eine Beschreibung der Skript-Testautomatisierung (Abschn. 6.2.2, 2. Absatz).
- c) FALSCH: Das ist eine Beschreibung der datengetriebenen Testautomatisierung (Abschn. 6.2.2, 3. Absatz).
- d) FALSCH: Dies beschreibt einen Teil dessen, was ein Testautomatisierungsrahmen oder ein Vergleichswerkzeug/Komparator tut (Abschn. 6.1.6).

39. Welche der folgenden Aussagen ist KEIN Ziel eines Pilotprojekts zur Werkzeug-Evaluierung? [K1]

a)	Beurteilen, wie das Werkzeug mit den bestehenden Prozessen und Praktiken zusammenpasst.	<input type="checkbox"/>
b)	Feststellen wie das Werkzeug und die damit verwalteten Testmittel verwendet, verwaltet, gespeichert und archiviert sowie gewartet werden.	<input type="checkbox"/>
c)	Beurteilen, ob der Nutzen bei vertretbaren Kosten erreicht wird.	<input type="checkbox"/>
d)	Reduzieren der Fehlerrate im Pilotprojekt.	<input type="checkbox"/>

Begründung

a) FALSCH: Der Satz ist WAHR (Abschnitt 6.3, Ziele, 2. dot)

b) FALSCH: Der Satz ist WAHR (Abschnitt 6.3, Ziele, 3. dot)

c) FALSCH: Der Satz ist WAHR (Abschnitt 6.3, Ziele, 4. dot)

d) KORREKT: Verringerung der Zahl der Mängel ist nicht das Ziel eines Pilotprojekts (vgl. Abschn. 6.3, Ziele).

40. Unten finden Sie eine Liste möglicher Ziele zur Effizienzsteigerung einer Software-Entwicklungs- und Testorganisation.

Welches dieser Ziele wird AM BESTEN durch ein Testmanagementwerkzeug unterstützt? [K2]

a)	Die Rückverfolgbarkeit zwischen Anforderungen, Testfällen und Fehlerzuständen herstellen.	<input type="checkbox"/>
b)	Die Fähigkeit der Tests optimieren, Fehlerwirkungen zu identifizieren.	<input type="checkbox"/>
c)	Die Fehlerzustände schneller beheben.	<input type="checkbox"/>
d)	Die Auswahl von Testfällen für die Testausführung automatisieren.	<input type="checkbox"/>

Begründung

- a) **KORREKT** - denn die Rückverfolgbarkeit zwischen Anforderungen und Tests ist eine Funktionalität eines Testmanagementwerkzeugs (Abschn. 6.1.3, Testmanagementwerkzeuge, Satz 1).
- b) **FALSCH** - weil das nicht mit Testmanagement-Werkzeugen möglich ist, eher durch Vergleichswerkzeuge/Komparatoren (Abschn. 6.1.6).
- c) **FALSCH** - weil das nicht in erster Linie durch Testmanagement-Werkzeuge gelöst wird, sondern nur ein wenig durch Rückverfolgbarkeit unterstützt wird (Abschn. 6.1.3, Testmanagementwerkzeuge, Satz 2).
- d) **FALSCH** - weil die Auswahl der Testfälle nicht durch Testmanagementwerkzeuge unterstützt wird, sondern durch Testentwurfs- und Testausführungswerkzeuge (Abschn. 6.1.5 und 6.1.6).