

# **Ejemplo de Examen**

## **Probador Certificado**

### **Nivel Básico del ISTQB®**

#### **Programa de Estudio 2011**

Versión ES.2.9.01

---

International Software Testing Qualifications Board

---



Nota sobre derechos de propiedad intelectual.

El presente documento podrá ser copiado parcial o íntegramente siempre y cuando se cite la fuente.

Copyright © 2017 International Software Testing Qualifications Board (en adelante denominado ISTQB™). Todos los derechos reservados.

Los autores transfieren los derechos de autor al International Software Testing Qualifications Board (en adelante denominado ISTQB™). Los autores (como titulares actuales de los derechos de autor) y el ISTQB™ (como el futuro titular de los derechos de autor) han acordado las siguientes condiciones de uso:

1) Cualquier Comité Miembro del ISTQB™ puede traducir este documento. Grupo de Trabajo de Examen ("Exam Working Party"): 2010 – 2017

## Control de Cambio

Versión	Fecha	Descripción
ES.2.9.01	30/01/2018	Traducción del documento "Sample Exam, ISTQB® Foundation Level, 2011 Syllabus, Version 2.9". Publicado el 16 de diciembre de 2017.

## Índice General

Control de Cambio.....	3
Índice General .....	4
Agradecimientos.....	6
0. Introducción .....	7
0.1 Objetivo de este Documento.....	7
0.2 Instrucciones .....	7
0.3 Información General sobre el Ejemplo de Examen .....	7
1. Preguntas "Principios Básicos del Proceso de Prueba".....	8
1.1 Pregunta 01.....	8
1.2 Pregunta 02.....	9
1.3 Pregunta 03.....	9
1.4 Pregunta 04.....	10
1.5 Pregunta 05.....	11
1.6 Pregunta 06.....	11
1.7 Pregunta 07.....	12
2. Preguntas "Pruebas durante todo el Ciclo de Vida del Software".....	14
2.0 Pregunta 08.....	14
2.1 Pregunta 09.....	14
2.2 Pregunta 10.....	15
2.3 Pregunta 11.....	16
2.4 Pregunta 12.....	16
2.5 Pregunta 13.....	17
3. Preguntas "Técnicas Estáticas".....	19
3.0 Pregunta 14.....	19
3.1 Pregunta 15.....	19
3.2 Pregunta 16.....	20
4. Preguntas "Técnicas de Diseño de Pruebas".....	21
4.0 Pregunta 17.....	21
4.1 Pregunta 18.....	22
4.2 Pregunta 19.....	24
4.3 Pregunta 20.....	25
4.4 Pregunta 21.....	26
4.5 Pregunta 22.....	27
4.6 Pregunta 23.....	28
4.7 Pregunta 24.....	29
4.8 Pregunta 25.....	29
4.9 Pregunta 26.....	30
4.10 Pregunta 27.....	31
4.11 Pregunta 28.....	32
5. Preguntas "Gestión de Pruebas".....	33
5.0 Pregunta 29.....	33

5.1	Pregunta 30.....	33
5.2	Pregunta 31.....	34
5.3	Pregunta 32.....	35
5.4	Pregunta 33.....	36
5.5	Pregunta 34.....	36
5.6	Pregunta 35.....	37
5.7	Pregunta 36.....	38
6.	Preguntas "Herramientas de Soporte de Pruebas".....	39
6.0	Pregunta 37.....	39
6.1	Pregunta 38.....	40
6.2	Pregunta 39.....	41
6.3	Pregunta 40.....	41

## Agradecimientos

Este documento ha sido traducido por el Spanish Software Testing Qualifications Board (SSTQB).

Este documento ha sido traducido por Gustavo Márquez Sosa (SSTQB).

## 0. Introducción

### 0.1 Objetivo de este Documento

Este documento contiene un ejemplo del examen completo siguiendo las reglas descritas en el documento "ISTQB Foundation Level Exam Structure and Rules"<sup>1</sup> (pendiente de traducir a fecha de publicación del presente documento).

Las preguntas, los conjuntos de respuestas y las justificaciones asociadas de este documento han sido creados por un equipo de expertos en la materia y redactores de preguntas experimentados con el objetivo de ayudar a los Comités Miembros del ISTQB™ y Comités de Examen en lo que respecta a la actividad de redacción de preguntas, así como a las personas que tienen previsto realizar el examen de "Probador Certificado - Nivel Básico" del ISTQB.

Estas preguntas no pueden ser utilizadas de forma literal en ningún examen oficial, pero deberían servir como orientación para los redactores de preguntas. Dada la amplia variedad de formatos y temas, estos ejemplos de preguntas deben ofrecer muchas ideas para los distintos Comités Miembros sobre cómo crear buenas preguntas y el conjunto de respuestas adecuado para sus exámenes. Además los proveedores de formación pueden utilizar estas preguntas como parte de su formación, a fin de preparar a los participantes para el examen.

### 0.2 Instrucciones

Los juegos de preguntas y respuestas están organizadas de la siguiente manera:

- Objetivo de Aprendizaje y nivel K
- Pregunta - incluyendo cualquier escenario seguido de la pregunta
- Conjunto de Respuestas
- Respuesta VERDADERA - incluyendo la justificación de las respuestas (la respuesta VERDADERA y las justificaciones se encuentran en un documento independiente).

### 0.3 Información General sobre el Ejemplo de Examen

Número de preguntas: **40**

Duración Estándar del Examen: **60 minutos**

Número total de puntos posibles: **40**

Puntuación para obtener el aprobado: **26 (o más)**

<sup>1</sup> Pendiente de traducir a fecha de publicación de la versión de este documento.

## 1. Preguntas "Principios Básicos del Proceso de Prueba"

### 1.1 Pregunta 01

OA-1.3.1	K2
----------	----

¿Cuál de las siguientes afirmaciones describe MEJOR uno de los siete principios básicos de las pruebas de software?

Juego de Respuestas:

- a) Es posible probar todo utilizando pruebas automatizadas.
- b) Las pruebas exhaustivas son posibles para todo software con un esfuerzo suficiente y herramientas de soporte.
- c) Normalmente es imposible probar todas las combinaciones entrada/salida para un sistema software.**
- d) El objetivo de las pruebas es para demostrar la ausencia de defectos.

Justificación:

- a) FALSO - La prueba exhaustiva es imposible, independientemente de que sea manual o automatizada. (Sección 1.3, Principio #2).
- b) FALSO - no es posible realizar pruebas exhaustivas, independientemente de la cantidad de esfuerzo dedicado a probar (Sección 1.3, Principio #2).
- c) VERDADERO - Principio #2 (Sección 1.3) establece: "Probar todo (todas las combinaciones de entradas y precondiciones) es imposible, salvo en casos triviales".**
- d) FALSO - Esta afirmación está en contradicción con el Principio #1 (Sección 1.3): Las pruebas demuestran la presencia de defectos: Las pruebas pueden demostrar que hay defectos, pero no pueden probar que no los hay.



1.2 Pregunta 02

OA-1.2.1	K1
----------	----

¿Cuál de las siguientes afirmaciones es la meta MÁS válida para un equipo de pruebas?

Juego de Respuestas:

- a) Para determinar si se han ejecutado suficientes pruebas de componente en las pruebas de sistema.
- b) Para detectar tantos fallos como sea posible, de manera que los defectos pueden ser identificados y corregidos.**
- c) Para demostrar que todos los posibles defectos son identificados.
- d) Para probar que ninguno de los defectos restantes causará ningún fallo.

Justificación:

- a) FALSO - Las pruebas de componente no forma parte de las pruebas de sistema. (Sección 2.2.1 y 2.2.3).
- b) VERDADERO - Esta es el rol principal de un equipo de pruebas. (Sección 1.2, objetivos, 1º punto).**
- c) FALSO - El Principio #1 establece que las pruebas exhaustivas no son posibles, por lo cual nunca se puede demostrar que todos los defectos fueron identificados.
- d) FALSO - Para hacer una valoración de si un defecto provocará un fallo o no, primero se tiene que haber detectado el defecto. Diciendo que ningún defecto restante provocará un fallo, significa, implícitamente, que se han detectado todos los defectos. Esta afirmación contradice el Principio #1.

1.3 Pregunta 03

OA-1.4.1	K1
----------	----

¿Cuál de estas tareas se puede esperar que se realicen durante la fase de Análisis y Diseño de Prueba del Proceso Básico de Prueba?

Juego de Respuestas:

- a) Definición de los objetivos de la prueba.
- b) Revisión de las bases de prueba.**
- c) Creación de juegos de prueba a partir de los procedimientos de prueba.
- d) Analizar las lecciones aprendidas para la mejora del proceso.

Justificación:

- a) FALSO - esta actividad se realiza durante la fase de "Planificación de la Prueba" (sección 1.4.1, frase 1).
- b) VERDADERO - esta actividad se realiza durante la fase de "Análisis y Diseño de Pruebas" (sección 1.4.2, punto 1º).**
- c) FALSO - esta actividad se realiza durante la fase de "Implementación y Ejecución de Pruebas" (sección 1.4.3, punto 3º).
- d) FALSO - esta actividad se realiza durante las "Actividades de Cierre de Pruebas" (sección 1.4.5, punto 6º).

<b>1.4 Pregunta 04</b>
------------------------

OA-1.1.5	K2
----------	----

A continuación se muestra una lista de problemas que se pueden observar durante la prueba o en producción. ¿Cuál de estos problemas es un fallo?

Juego de Respuestas:

- a) El producto "dejó de funcionar" ("crashed") cuando el usuario seleccionó una opción en un cuadro de diálogo.**
- b) Un archivo de código fuente incluido en la compilación tiene la versión FALSA.
- c) El algoritmo de cálculo ha utilizado variables de entrada FALSAS.
- d) El desarrollador ha malinterpretado el requisito para el algoritmo.

Justificación:

- a) VERDADERO - Un fallo es una manifestación externa de un defecto. Un fallo de producto ("crash") es claramente perceptible por un usuario.**
- b) FALSO - Este tipo de equivocación no conducirán necesariamente a fallo visible o notorio. Por ejemplo: si los cambios en la nueva versión del archivo de código fuente se encuentran en los comentarios.
- c) FALSO - El uso de una variable de entrada FALSA no conducirá, necesariamente, a un fallo visible o notorio. Por ejemplo: si nadie utiliza este algoritmo específico, o si la variable de entrada FALSA utilizada tenía un valor similar al de la variable de entrada VERDADERA, o si nadie está utilizando el resultado FALSO del algoritmo. "Defectos de software, sistemas o documentos pueden resultar en fallos, pero no todos los defectos lo hacen". (Sección 1.1.2, Párrafo 1º, punto 3º).
- d) FALSO - Este tipo de equivocación no conducirá necesariamente a un fallo visible o notorio. Por ejemplo: si nadie utiliza este algoritmo específico.

1.5 Pregunta 05

OA-1.5.1	K1
----------	----

¿Cuál de las siguientes actitudes, conocimientos o acciones llevaría a problemas (o conflictos) en equipos mixtos de los desarrolladores y probadores de software, cuando se observa en las revisiones y pruebas?

Juego de Respuestas:

- a) Los desarrolladores y probadores de software son curiosos y se concentran en encontrar defectos.
- b) Los desarrolladores y probadores de software están suficientemente cualificados para encontrar fallos y defectos.
- c) Los desarrolladores y probadores de software comunican defectos como una crítica personal, no como una crítica al producto software.
- d) Los probadores esperan que podría haber defectos en el producto software que los desarrolladores no han encontrado y solucionado.

Justificación:

- a) FALSO. No hay ninguna situación que conduzca a un conflicto. Los desarrolladores y probadores de software deben estar concentrados en la búsqueda de defectos.
- b) FALSO. Esta es una situación ideal, por lo tanto, no hay ningún problema.
- c) **VERDADERO. De acuerdo al programa de estudio, los desarrolladores y probadores de software deben cooperar, y comunicar los defectos como una crítica personal podría conducir a conflictos dentro del equipo. (Sección 1.5.9, Párrafo 9º, punto 2º punto).**
- d) FALSO. El papel del probador en el equipo es encontrar defectos en el producto software que los desarrolladores no han encontrado y solucionado.

1.6 Pregunta 06

OA-1.2.2	K2
----------	----

¿Cuál de las siguientes afirmaciones es VERDADERA?

- I. Las pruebas de software pueden ser necesarias para satisfacer requisitos legales o contractuales.
- II. Las pruebas de software son necesarias, principalmente, para mejorar la calidad del producto.
- III. Unas pruebas y corrección de defectos rigurosas ayudarían a reducir el riesgo de que se produjeran problemas en un entorno de operaciones.
- IV. Unas pruebas rigurosas se utilizan, en ocasiones, para demostrar de que todos los fallos han sido encontrados.

### Juego de Respuestas:

- a) **I, II y III son verdaderas; IV es falsa.**
- b) I es verdadera; II, III y IV son falsas.
- c) I y III son verdaderas; II y IV son falsas.
- d) III y IV son verdaderas; I y II son falsas.

### Justificación:

I. VERDADERO. Las pruebas de software pueden ser necesarias para satisfacer requisitos legales o contractuales. (Sección 2.2.4)

II. VERDADERO. Las pruebas de software son necesarias, principalmente, para mejorar la calidad del producto. (LO 1.1.4, Sección 1.1.3, Párrafo 1º, Sección 1.1.4, Sección 2ª Sección 1.5, última enumeración, punto 1º)

III. VERDADERO. Uno de los principales objetivos de las pruebas de software es reducir el riesgo de que se produzcan problemas en un entorno de operaciones. (Sección 1.2, objetivos 1 y 4)

IV. FALSO. Es imposible probar que todos los fallos han sido encontrados. (Sección. 1.3, principio 2)

Por lo tanto

- a) **VERDADERO, porque I, II y III son VERDADERAS.**
- b) FALSO, porque II y III NO SON VERDADERAS.
- c) FALSO, porque II NO ES FALSA.
- d) FALSO, porque IV NO ES VERDADERA y porque I y II SON VERDADERAS.

<b>1.7 Pregunta 07</b>
------------------------

OA-1.2.3	K2
----------	----

¿Cuál de las siguientes afirmaciones describe correctamente la diferencia entre las pruebas y la depuración?

### Juego de Respuestas:

- a) Las pruebas identifican el origen de los defectos, la depuración analiza los defectos y propone actividades de prevención.
- b) **Las pruebas dinámicas muestran los fallos causados por defectos, la depuración encuentra, analiza y elimina las causas de fallos en el software.**
- c) Las pruebas eliminan los defectos, la depuración identifica las causas de los fallos.
- d) Las pruebas dinámicas impide las causas de fallos, la depuración elimina los fallos.

Justificación:

- a) FALSO. Las pruebas no identifican el origen de los defectos (Sección 1.2).
- b) VERDADERO. Las pruebas dinámicas muestran fallos causados por defectos, la depuración encuentra, analiza y elimina las causas de los fallos en el software (Sección 1.2).**
- c) FALSO. La prueba no elimina los fallos (Sección 1.2).
- d) FALSO. Las pruebas dinámicas no impiden las causas de fallos (Sección 1.3).

## 2. Preguntas "Pruebas durante todo el Ciclo de Vida del Software"

### 2.0 Pregunta 08

OA-2.3.3	K1
----------	----

¿Cuál de las siguientes afirmaciones describe MEJOR las pruebas no funcionales?

Juego de Respuestas:

- a) Las pruebas no funcionales son el proceso de prueba de un sistema integrado para verificar que cumple con los requisitos especificados.
- b) Las pruebas no funcionales son el proceso de pruebas para determinar la conformidad del sistema con los estándares de codificación.
- c) Las pruebas no funcionales son probar sin hacer referencia a la estructura interna del sistema.
- d) Las pruebas no funcionales son probar características del sistema, tales como la usabilidad, fiabilidad o mantenibilidad.**

Justificación:

- a) FALSO, esto es una definición de pruebas de sistema (Sección 2.2.3).
- b) FALSO, esto es una función del análisis estático soportado por una herramienta (Sección 3.3 y Sección 6.1.4).
- c) FALSO, es una definición de pruebas de caja negra (Sección 4.2).
- d) VERDADERO, las pruebas de características del sistema, tales como la usabilidad, fiabilidad o mantenibilidad son pruebas no funcionales (Sección 2.3.2).**

### 2.1 Pregunta 09

OA-2.1.x	K1
----------	----

¿Qué es importante hacer cuando se trabaja con modelos de desarrollo de software?

Juego de Respuestas:

- a) Si fuera necesario, adaptar los modelos a las características del proyecto y del producto.**
- b) Elegir el modelo en cascada, porque es el modelo más consolidado.
- c) Comience con el modelo-V y, a continuación, pasar al modelo iterativo o al incremental.
- d) Cambiar la organización que se adapte al modelo, y no al revés.

Justificación:

- a) **VERDADERO - Los modelos aportan directrices generales - no aportan un proceso preciso, paso a paso, que deba ser seguido de forma literal. (Sección 2.1).**
- b) FALSO - El modelo en cascada sólo es uno de los posibles modelos a los que un equipo puede optar por seguir.
- c) FALSO - El modelo-V (Sección 2.1.1) no es compatible con los modelos iterativos. (Sección 2.1.2) de manera que el flujo descrito no tiene sentido.
- d) FALSO - Los modelos escogidos para adaptarse a la situación y al proyecto y no al revés (Sección 2.1, último Párrafo).

<b>2.2 Pregunta 10</b>
------------------------

OA-2.1.3	K1
----------	----

¿Cuál de las siguientes afirmaciones es una característica de buena prueba y se aplica en cualquier modelo de ciclo de vida de desarrollo de software?

Juego de Respuestas:

- a) La prueba de aceptación siempre es el último nivel de la prueba que se aplica.
- b) Todos los niveles de prueba son planificados y completados para cada característica desarrollada.
- c) Los probadores son involucrados por primera vez cuando la primera pieza de código puede ser ejecutada.
- d) **Para cada actividad de desarrollo hay una actividad de prueba correspondiente.**

Justificación:

- a) FALSO - Esto sólo es VERDADERO para los proyectos que tienen pruebas de aceptación. Algunos proyectos no tienen este nivel de prueba. (Véase Sección 2.1).
- b) FALSO - Hay casos donde algunos niveles de prueba no son estrictamente necesarios. Por ejemplo: al obtener código de un tercero, las pruebas de componente no son necesarias.
- c) FALSO - Los probadores deben participar mucho antes de que el código esté disponible. Por ejemplo, los probadores deberían estar involucrados en las revisiones de la especificación de requisitos. (Sección 1.4.2).
- d) **VERDADERO - "En cualquier modelo de ciclo de vida se dan varias características de buenas pruebas: Para cada actividad de desarrollo existe una actividad de prueba correspondiente". (Sección 2.1.3).**

**2.3 Pregunta 11**

OA-2.4.1	K1
----------	----

¿Cuál de las siguientes descripciones es un ejemplo de prueba de mantenimiento?

Juego de Respuestas:

- a) Para probar defectos corregidos durante el desarrollo de un nuevo sistema.
- b) Para probar mejoras en un sistema existente en explotación.
- c) Para tratar las quejas con respecto a la calidad del sistema durante la prueba de aceptación de usuario.
- d) Para integrar las funciones durante el desarrollo de un nuevo sistema.

Justificación:

- a) FALSO - Probar un nuevo sistema no es "prueba de mantenimiento" (Sección 2.4).
- b) VERDADERO - probar la capacidad del sistema para desempeñar su función después de un cambio en el entorno se considera "prueba de mantenimiento". (Sección 2.4).**
- c) FALSO - Tratar los fallos provenientes de la prueba de aceptación no es "prueba de mantenimiento" (véase también Sección 2.4).
- d) FALSO - La integración de funciones no es una actividad de pruebas (véase también Sección 2.2.2).

**2.4 Pregunta 12**

OA-2.3.5	K2
----------	----

¿Cuál de las siguientes afirmaciones es VERDADERA?

- I. Las pruebas de regresión y de pruebas de confirmación son lo mismo.
- II. Las pruebas de regresión demuestran que todos los defectos se han resuelto.
- III. La prueba de regresión es un buen candidato para la automatización de prueba.
- IV. La prueba de regresión se realiza para descubrir defectos como resultado de los cambios en el software.
- V. No se deberían realizar pruebas de regresión durante las pruebas de integración.

Juego de Respuestas:

- a) IV es VERDADERA; I, II, III y V son FALSAS.
- b) III es VERDADERA; I, II, IV y V son FALSAS.
- c) III y IV son VERDADERAS; I, II y V son FALSAS.**
- d) I, III y IV son VERDADERAS; II y V son FALSAS.



### Justificación:

La afirmación I es FALSA - la prueba de regresión es la repetición de la prueba de un programa ya probado, después de la modificación, para descubrir los defectos introducidos o descubiertos como resultado del cambio (uno o más cambios).

Las pruebas de confirmación se realizan para confirmar que un defecto se ha eliminado de forma correcta. (Sección 2.3.4).

La afirmación II es FALSA - la frase describe las pruebas de confirmación. (Sección 2.3.4).

La afirmación III es VERDADERA - los juegos de pruebas de regresión se ejecutan muchas veces y, en general, evolucionan lentamente, de manera que la prueba de regresión es un fuerte candidato para la automatización. (Sección 2.3.4).

La afirmación IV es VERDADERA - esta es la definición de pruebas de regresión (véase Sección 2.3.4).

La afirmación V es FALSA - "la prueba de regresión puede realizarse en todos los niveles de prueba, e incluye a las pruebas funcionales, no funcionales y estructurales". (Sección 2.3.4).

Por lo tanto

- a) INCORRECTO, porque I y II son FALSAS.
- b) INCORRECTO, porque I y V son FALSAS.
- c) CORRECTO, porque III y IV son VERDADERAS.**
- d) INCORRECTO, porque II y V son FALSAS.

## 2.5 Pregunta 13

OA-2.2.1

K2

¿Cuál de las siguientes afirmaciones comparando las pruebas de componente y las pruebas de sistema es VERDADERA?

### Juego de Respuestas:

a) Las pruebas de componente verifican la funcionalidad de los módulos de software, objetos del programa y clases que pueden ser probadas por separado, mientras que las pruebas de sistema verifican las interfaces entre los componentes y las interacciones entre las diferentes partes del sistema.

**b) Los casos de prueba para las pruebas de componente, en general, se obtienen a partir de las especificaciones de componente, especificaciones de diseño o modelos de datos, mientras que los casos de prueba para las pruebas de sistema se suelen obtener a partir de especificaciones de requisitos, especificaciones funcionales, o casos de uso.**

c) Las pruebas de componente sólo se centran en características funcionales, mientras que las pruebas de sistema se centran en las características funcionales y no funcionales.

d) Las pruebas de componente son responsabilidad de los probadores, mientras que las pruebas de sistema, normalmente, son responsabilidad de los usuarios del sistema.

### Justificación:

- a) FALSO. Las pruebas de sistema no prueban las interfaces entre componentes y las interacciones entre diferentes partes del sistema, es un objetivo de pruebas de integración. (Véase también Sección 2.2.2).
- b) VERDADERO. Sección 2.2.1 (Pruebas de componente), base de prueba, 1. Dot; Sección 2.2.3 (Pruebas de Sistema) base de prueba, 1. - 3. Punto.
- c) FALSO, pruebas de componente no sólo se centran en características funcionales (véase también Sección 2.2.1, Párrafo 2º; después de "Objetos de prueba típicos").
- d) FALSO, las pruebas de componente son, normalmente, responsabilidad de los desarrolladores, mientras que las pruebas de sistema son, normalmente, responsabilidad de probadores (véase también Sección 2.2.3, 4. Párrafo después de "Objetos de prueba típicos", frase 2ª).

### 3. Preguntas "Técnicas Estáticas"

#### 3.0 Pregunta 14

OA-3.2.1

K1

¿Cuál de las siguientes afirmaciones describe las fases principales de una revisión formal?

Juego de Respuestas:

- a) Iniciación, retroceso, preparación individual, reunión de revisión, reconstrucción, seguimiento.
- b) Planificación, preparación individual, reunión de revisión, reconstrucción, cierre, seguimiento.
- c) Planificación, inicio, preparación individual, reunión de revisión, reconstrucción, seguimiento.**
- d) Preparación individual, reunión de revisión, reconstrucción, cierre, seguimiento, análisis de causa raíz.

Justificación:

Las fases principales de una revisión formal son la **planificación, inicio, preparación individual, reunión de revisión, reconstrucción y seguimiento**, por lo tanto:

- a) FALSO: Faltan planificación e inicio (véase también Sección 3.2.1), iniciación y retroceso (véase también Sección 3.2.1) no están incluidos en el proceso.
- b) FALSO: Falta inicio (véase la sección 3.2.1), el cierre no es parte del proceso de revisión.
- c) VERDADERO: (véase también Sección 3.2.1).**
- d) FALSO: Faltan planificación e inicio (véase también Sección 3.2.1), el cierre y el análisis de la causa raíz no son parte del proceso de revisión.

#### 3.1 Pregunta 15

OA-3.2.2

K2

¿Cuál de los siguientes tipos de revisión es la mejor opción para seleccionar con el objeto de llevar a cabo revisiones de componentes de seguridad crítica en un proyecto de software si, además, la revisión debe demostrarse como proceso formal basado en reglas y listas de comprobación?

Juego de Respuestas:

- a) Revisión Informal.
- b) Revisión Técnica.
- c) Inspección.
- d) Revisión Guiada.

Justificación:

Para revisar componentes de seguridad crítica en un proyecto software hace falta una revisión más formal, documentada, basada en reglas y listas de comprobación, por lo tanto se necesita una inspección.

Por lo tanto:

- a) FALSO
- b) FALSO
- c) VERDADERO**
- D) FALSO

**3.2 Pregunta 16**

OA-3.3.2	K1
----------	----

¿Cuál de las siguientes afirmaciones acerca del análisis estático basado en herramientas es FALSA?

Juego de Respuestas:

- a) El análisis estático basado en herramientas puede ser utilizado como una medida preventiva con procesos adecuados en su sitio.
- b) El análisis estático basado en herramientas pueden detectar defectos que no se encuentran con facilidad con pruebas dinámicas.
- c) El análisis estático basado en herramientas puede resultar en ahorro de costes por una detección temprana de defectos.
- d) El análisis estático basado en herramientas es una buena manera de forzar fallos en el software.**

Justificación:

- a) FALSO. Esta afirmación es verdadera, el análisis estático basado en herramientas puede ser utilizado como una medida preventiva (Sección 3.3, ventajas 2ª y 6ª).
- b) FALSO. Esta afirmación es verdadera, el análisis estático basado en herramientas puede "Identificación de defectos que no se encuentran fácilmente mediante pruebas dinámicas" (Sección 3.3, Antecedentes, Párrafo 1º, frase 3ª).
- c) FALSO. Esta afirmación es verdadera, el análisis estático basado en herramientas es un método de ahorro de costes utilizado para encontrar defectos de forma temprana (Sección 3.3, ventaja 1ª.)
- d) VERDADERO. Esta afirmación es incorrecta. Durante el análisis estático no se detectan fallos porque el código no se ejecuta (Sección 3.2, antecedentes, frase 2).**

## 4. Preguntas "Técnicas de Diseño de Pruebas"

### 4.0 Pregunta 17

OA-4.4.4

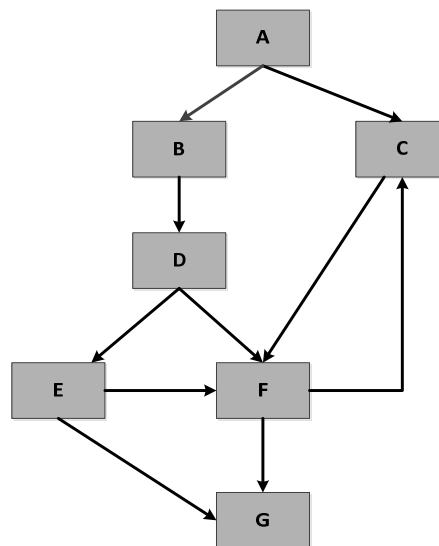
K4

Uno de los objetivos de su proyecto es tener un 100% de cobertura de decisión. Las tres pruebas siguientes han sido ejecutadas para el flujo de control del siguiente grafo (diagrama).

Prueba\_01 cubre el camino: A, B, D, E, G

Prueba\_02 cubre el camino: A, B, D, E, F, G

Prueba\_03 cubre el camino: A, C, F, C, F, C, F, G



¿Cuál de las siguientes afirmaciones en relación con el objetivo de cobertura de decisión es VERDADERA?

Juego de Respuestas:

- a) **La decisión D no se ha probado de forma completa.**
- b) Se ha logrado un 100% de cobertura de decisión.
- c) La decisión E no se ha probado de forma completa.
- d) La decisión F no se ha probado de forma completa.

### Justificación:

En el siguiente diagrama hay cuatro condiciones/decisiones: A, D, E, F

La Prueba\_01 cubre A->B, D->E y E->G

La Prueba\_02 cubre A->B, D->E, E->F y F->G

La Prueba\_03 cubre A->C, F->C y F->G.

Por lo tanto:

la condición A está cubierta (A->B por Prueba\_01 y A->C por Prueba\_03),

la condición E está cubierta (E->G por Prueba\_01 y E->F por Prueba\_02),

la condición F está cubierta (F->C por Prueba\_03 y F->G por Prueba\_02 y Prueba\_03).

la condición D no está cubierta, sólo están cubiertas D->E por Prueba\_01 y Prueba\_02, pero D->F no está cubierta.

Por lo tanto

**a) VERDADERO: D no estaba cubierto (ver arriba).**

b) FALSO: 1 de 4 decisiones no está cubierta (véase arriba), hay, tan sólo, una cobertura del 75%.

c) FALSO: E ha sido probada/cubierta de forma completa (véase arriba).

d) FALSO: F ha sido probada/cubierta de forma completa (véase arriba).

## 4.1 Pregunta 18

OA-4.2.2

K2

Se ha detectado un defecto durante la prueba:

Al recibir los datos del cliente desde un servidor, se produce un fallo del sistema. El defecto fue corregido al corregir el código que comprueba la disponibilidad de la red durante la transferencia de datos. Los casos de prueba existentes cubren el 100% de todas las sentencias del módulo correspondiente. Se han diseñado, agregado al juego de pruebas y ejecutado algunas pruebas adicionales con el objetivo de verificar la corrección y asegurar una cobertura más amplia.

¿Cuál de los siguientes tipos de deberían utilizar en este escenario?

I. Pruebas funcionales.

II. Pruebas estructurales.

III. Pruebas de confirmación o repetición de pruebas.

IV. Pruebas de rendimiento.

### Juego de Respuestas:

- a) Se utilizan I y II, pero no se utilizan III y IV.
- b) Se utilizan I y III, pero no se utilizan II y IV.
- c) Se utilizan I, II y III, pero no se utiliza IV.**
- d) Se utilizan II, III y IV, pero no se utiliza I.

### Justificación:

I. VERDADERO: Recibir datos del cliente es una prueba funcional típica.

II. VERDADERO: En la descripción del problema, se dijo, "Los casos de prueba existentes cubren el 100% de todas las sentencias del módulo correspondiente."; las pruebas de sentencia corresponden a pruebas estructurales.

III. VERDADERO: En la descripción del problema, se dijo, "Se han diseñado, agregado al juego de pruebas y ejecutado algunas pruebas adicionales con el objetivo de verificar la corrección y asegurar una cobertura más amplia." por lo tanto, esto es una nueva comprobación.

IV. FALSO. En la descripción del problema no hay información sobre pruebas de rendimiento.

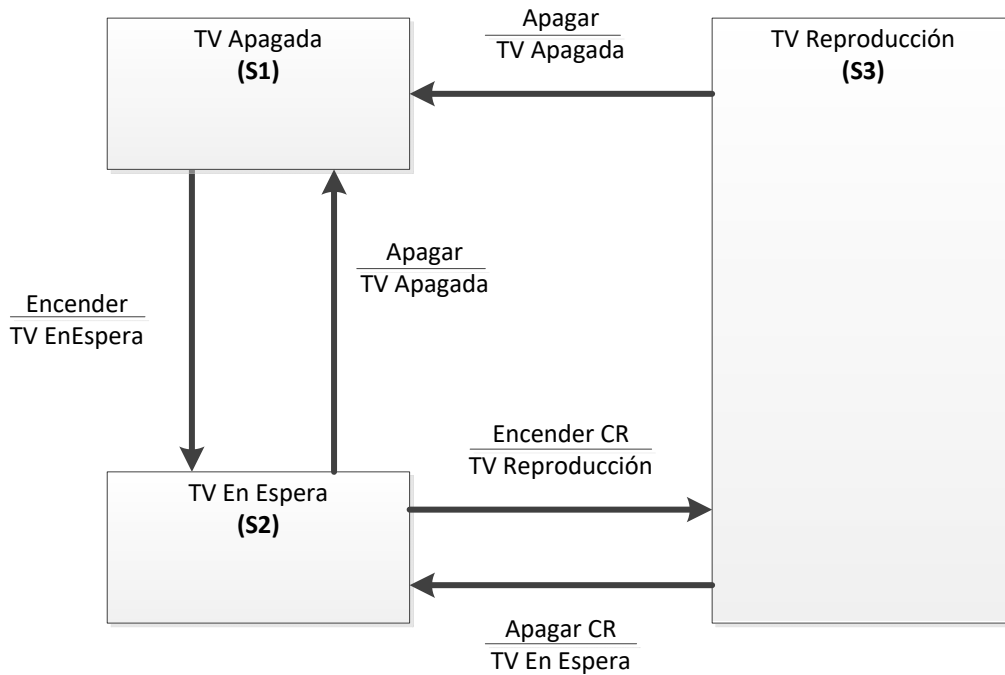
Por lo tanto

- a) FALSO, dado que falta III.
- b) FALSO, dado que falta II.
- c) VERDADERO, como I, II y III son mencionados.**
- d) FALSO, dado que falta I y IV se menciona erróneamente.

**4.2 Pregunta 19**

OA-4.3.1	K3
----------	----

¿Cuál de las siguientes afirmaciones sobre el diagrama de transición de estado y la tabla de casos de prueba es VERDADERA?



Dónde:

- **CR** es el acrónimo de Control Remoto.

Caso de Prueba	1	2	3	4	5
<b>Estado Inicial</b>	S1	S2	S2	S3	S3
<b>Entrada</b>	Encender	Apagar	Encender CR	Apagar CR	Apagar
<b>Salida Esperada</b>	TV En Espera	TV Apagada	TV Reproducción	TV En Espera	TV Apagada
<b>Estado Final</b>	S2	S1	S3	S2	S1



Juego de Respuestas:

- a) Los casos de prueba dados se pueden utilizar para obtener tanto las transiciones válidas como las transiciones inválidas en el diagrama de transición de estado.
- b) Los casos de prueba dados representan todas las posibles transiciones válidas en el diagrama de transición de estado.**
- c) Los casos de prueba dados sólo representan algunas de las transiciones válidas en el diagrama de transición de estado.
- d) Los casos de prueba dados representan pares de transiciones secuenciales en el diagrama de transición de estado.

Justificación:

Los casos de prueba propuestos cubren la totalidad de las cinco posibles transiciones unitarias válidas en el diagrama de transición de estado (S1->S2, S2->S1, S2->S3, S3->S2, S3->S1).

Por lo tanto:

- a) Es FALSO, porque no están cubiertas las transiciones inválidas.
- b) Es VERDADERO, porque todas las transiciones válidas están cubiertas.**
- c) Es FALSO, porque todas las transiciones válidas están cubiertas.
- d) Es FALSO, porque los casos de prueba no contienen pares de transiciones.

<b>4.3 Pregunta 20</b>
------------------------

OA-4.3.2	K2
----------	----

¿Cuál de las siguientes afirmaciones para la técnica de prueba de partición de equivalencia es VERDADERA?

Las técnicas de pruebas de partición de equivalencia...

- I. divide las posibles entradas en clases, donde se espera que todos los elementos causen el mismo comportamiento.
- II. utiliza ambas particiones, válidas e inválidas.
- III. debe incluir, por lo menos, dos valores de cada partición de equivalencia.
- IV. sólo se pueden utilizar para probar entradas de particiones de equivalencia desde una Interfaz Gráfica de Usuario.

Juego de Respuestas:

- a) I, II y IV son VERDADERAS; III es FALSA.
- b) I es VERDADERA; II, III y IV son FALSAS.
- c) II y III son VERDADERAS; I y IV son FALSAS.
- d) I y II son VERDADERAS; III y IV son FALSAS.**

Justificación:

- I. Es VERDADERA, dado que las particiones de equivalencia dividen las posibles entradas en clases donde se espera que todos los elementos provoquen el mismo comportamiento (Sección 4.3.1, frase 1ª).
- II. Es VERDADERA, ya que las particiones de equivalencia utilizan ambas particiones, válidas e inválidas. (Sección 4.3.1, frase 2ª).
- III. Es FALSA, es suficiente utilizar un valor de cada clase. (Sección 4.3.1, Párrafo 1º).
- IV. Es FALSA, es posible introducir una entrada desde la interfaz gráfica de usuario ("GUI"), pero este método no es necesario.

Por lo tanto

- a) FALSA, porque IV no es VERDADERA.
- b) FALSA, porque II NO ES FALSA.
- c) FALSA, porque III NO ES VERDADERA y I NO ES FALSA.
- d) **VERDADERA, porque I y II VERDADERAS y III y IV son FALSAS.**

4.4 Pregunta 21
-----------------

OA-4.2.2	K1
----------	----

¿Cuál de las siguientes opciones enumeran técnicas clasificadas como técnicas de diseño de Caja Negra?

Juego de Respuestas:

- a) Particiones de Equivalencia, Pruebas de Tabla de Decisión, Pruebas de Transición de Estado, Análisis de Valores Frontera.
- b) Particiones de Equivalencia, Pruebas de Tabla de Decisión, Cobertura de Sentencia, Pruebas Basadas en Casos de Uso.
- c) Particiones de Equivalencia, Pruebas de Cobertura de Decisión, Pruebas Basadas en Casos de Uso.
- d) Particiones de Equivalencia, Pruebas de Cobertura de Decisión, Análisis de Valores Frontera.

Justificación:

- a) VERDADERO - las cuatro son técnicas de prueba de caja negra. (Véase Sección 4.3).
- b) FALSO - Cobertura de Sentencia es una técnica de prueba de caja blanca (Sección 4.4.1).
- c) FALSO - Cobertura de Decisión es una técnica de prueba de caja blanca (Sección 4.4.2).
- d) FALSO - Cobertura de Decisión es una técnica de prueba de caja blanca (Sección 4.4.2).

<b>4.5 Pregunta 22</b>
------------------------

OA-4.3.1	K3
----------	----

Se debe calcular la bonificación de un empleado. No puede tener un valor negativo, pero el resultado del cálculo puede ser cero. La bonificación se basa en la antigüedad del empleado.

Las categorías son: inferior o igual a 2 años, más de 2 años, pero menos de 5 años, 5 años o más, pero menos de 10 años, 10 años o más. Dependiendo de la antigüedad, el empleado obtendrá diferentes niveles de bonificación.

¿Cuántos casos de prueba son necesarios, si sólo se necesitan particiones de equivalencia válidas para probar el cálculo de la bonificación?

Juego de Respuestas:

- a) 3
- b) 5
- c) 2
- d) 4**

Justificación:

- a) FALSO - vea las particiones VERDADERAS en d).
- b) FALSO - vea las particiones VERDADERAS en d).
- c) FALSO - vea las particiones VERDADERAS en d).
- d) VERDADERO. Las particiones siguen la descripción de la pregunta:**
  - 1. partición de equivalencia:  $0 < \text{antigüedad} \leq 2$**
  - 2. partición de equivalencia:  $2 < \text{antigüedad} < 5$**
  - 3. partición de equivalencia:  $5 \leq \text{antigüedad} < 10$**
  - 4. partición de equivalencia:  $10 \leq \text{antigüedad}$**

<b>4.6 Pregunta 23</b>
------------------------

OA-4.3.5	K3
----------	----

¿Cuáles de las siguientes afirmaciones acerca de los beneficios de generar casos de prueba a partir de casos de uso son verdaderas y cuáles son falsas?

- I. Derivar casos de prueba de los casos de uso es útil para las pruebas de sistema y pruebas aceptación.
- II. Derivar casos de prueba de casos de uso es útil sólo para pruebas automatizadas.
- III. Derivar casos de prueba de casos de uso es útil para las pruebas de componente.
- IV. Derivar casos de prueba de los casos de uso es útil para las pruebas de integración.

Juego de Respuestas:

- a) **I y IV son VERDADERAS; II y III son FALSAS.**
- b) I, es VERDADERA; II, III y IV son FALSAS.
- c) II y IV son VERDADERAS; I y III son FALSAS.
- d) I, III y IV son VERDADERAS; II es FALSA.

Justificación:

La afirmación I es VERDADERA - Los casos de uso describen cómo interactúan los usuarios con el sistema completado, por lo que se adaptan mejor para definir las pruebas a nivel de sistema. Además, "Los casos de uso son de gran utilidad para diseñar pruebas de aceptación con la participación del cliente/usuario." (Sección 4.3.5).

La afirmación II es FALSA - Los casos de uso se pueden ejecutar manualmente, no sólo automáticamente.

La afirmación III es FALSA - A nivel de componente derivamos casos de prueba para aumentar la cobertura de código. Éstos se derivan, en su mayoría, mirando el código, no por definición de un caso de uso que eventualmente practicará una cierta parte del código. (Sección 2.2.1).

La afirmación IV es VERDADERA - "[los casos de uso] también ayudan a descubrir defectos de integración causados por la interacción e interferencia de diferentes componentes" (Sección 4.3.5).

Por lo tanto

- a) **VERDADERA, porque I y IV son VERDADERAS.**
- b) FALSA, porque IV es VERDADERA (y NO FALSA).
- c) FALSA, porque II es FALSA (y NO ES VERDADERA) y I VERDADERA (y NO ES FALSA).
- d) FALSA, porque III es FALSA (y NO ES VERDADERA).

**4.7 Pregunta 24**

OA-4.5.1

K2

¿Cuál de las siguientes opciones sería la MEJOR base para probar con ataques de falta?

Juego de Respuestas:

**a) Experiencia, datos de defectos y fallos, conocimiento sobre fallos del software.**

b) Identificación de riesgos realizada al inicio del proyecto.

c) Casos de Uso derivados de los flujos de negocio por expertos del dominio.

d) Resultados esperados de la comparación con un sistema existente.

Justificación:

**a) VERDADERA - "Esta lista de defectos y fallos puede elaborarse en base a la experiencia, a los datos disponibles sobre defectos y fallos y a partir del conocimiento común sobre por qué falla el software.". (Sección 4.5, Párrafo 2º, última frase).**

b) FALSA - La identificación del riesgo marca las áreas o características de interés en el proyecto - no cómo probarlas.

c) FALSA - La prueba de los flujos de negocio no está dirigida a las debilidades conocidas del software. Sólo intenta verificar que ciertos casos de uso pueden ser ejecutados. (Sección 4.5, Párrafo 2º, última frase).

d) FALSA - La oración describe una de las posibles maneras de saber si una prueba falló o no y no tiene nada específico que ver con los ataques de faltas. (Sección 4.5 2, Párrafo 2º, última frase).

**4.8 Pregunta 25**

OA-4.5.1

K1

Usted está trabajando en un proyecto que tiene unas malas especificaciones y presión en lo que respecta al tiempo. ¿Cuál de las siguientes técnicas de prueba sería el enfoque más útil a utilizar?

Juego de Respuestas:

a) Pruebas de Casos de Uso.

b) Pruebas de Sentencia.

**c) Pruebas Exploratorias.**

d) Pruebas de Decisión.

### Justificación:

- a) FALSO, el proyecto tiene malas especificaciones, por lo que sólo existe una pequeña posibilidad de que existan casos de uso.
- b) FALSO, las pruebas de sentencia consumen mucho tiempo, y hay presión en lo que respecta al tiempo en el proyecto.
- c) **VERDADERO, las pruebas exploratorias son una buena idea cuando hay poca documentación y presión en lo que respecta al tiempo (Sección 4.5, Introducción, Párrafo 3º, Frase 3ª).**
- d) FALSO, las pruebas de decisión consumen mucho tiempo, y hay una presión en lo que respecta al tiempo en el proyecto.

### 4.9 Pregunta 26

OA-4.2.2	K1
----------	----

¿Cuál de las siguientes técnicas de prueba es una técnica de caja blanca?

### Juego de Respuestas:

- a) Pruebas de Decisión.
- b) Análisis de Valores Frontera.
- c) Particiones de Equivalencia.
- d) Pruebas de Transición de Estado.

### Justificación:

- a) **VERDADERA, la prueba de decisión es una técnica de caja blanca (Sección 4.4.2).**
- b) FALSO, el Análisis del Valores Frontera es una técnica de caja negra (Sección 4.3.2).
- c) FALSO, la Partición de Equivalencia es una técnica de caja negra (Sección 4.3.1).
- d) FALSO, la prueba de Transición de Estado es una técnica de caja negra (Sección 4.3.4).

**4.10 Pregunta 27**

OA-4.3.1	K3
----------	----

Usted está probando un sistema que calcula el Máximo Común Divisor (MCD) de dos números enteros (A y B) mayores que cero. [K3]

calcMCD (A, B);

Se han especificado las siguientes entradas de prueba.

Caso de prueba	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

Dónde INT\_MAX es el mayor número entero.

¿Qué técnica de prueba basada en especificaciones utilizaría para los casos de prueba 1 a 6?

Juego de Respuestas:

- a) **Análisis de Valores Frontera.**
- b) Pruebas de Transición de Estado.
- c) Pruebas de Casos de Uso.
- d) Pruebas de Tabla de Decisión.

Justificación:

- a) **VERDADERO: dados los valores (0,1,INT\_MAX+1, INT\_MAX) son valores frontera típicos.**
- b) FALSO, véase a) para su justificación.
- c) FALSO, véase a) para su justificación.
- d) FALSO, véase a) para su justificación.

**4.11 Pregunta 28**

OA-4.3.1	K3
----------	----

Los empleados de una empresa reciben primas si trabajan más de un año en la empresa y alcanzan los objetivos acordados individualmente.

La siguiente tabla de decisiones ha sido diseñada para probar el sistema:

		T1	T2	T3	T4	T5	T6	T7	T8
<b>Condiciones</b>									
Cond1	¿Empleo por más de 1 año?	SI	NO	SI	NO	SI	NO	SI	NO
Cond2	¿Objetivo acordado?	NO	NO	SI	SI	NO	NO	SI	SI
Cond3	¿Objetivo alcanzado?	NO	NO	NO	NO	SI	SI	SI	SI
<b>Acción</b>									
	¿Pago de bono?	NO	NO	NO	NO	NO	NO	SI	NO

¿Qué casos de prueba que podrían ser eliminados de la tabla de decisión anterior debido a que el caso de prueba no podría ocurrir en una situación real?

Juego de Respuestas:

- a) T1 y T2
- b) T3 y T4
- c) T7 y T8
- d) T5 y T6**

Justificación:

En los casos de prueba uno debe inferir a partir de las condiciones. En los casos de prueba T5 y T6 se describe la situación, pero no se llegó a un acuerdo sobre el objetivo. Debido a que esta situación no puede ocurrir, podemos eliminar los casos de prueba correspondientes.

Por lo tanto

- a) FALSO
- b) FALSO
- c) FALSO
- D) VERDADERO**



## 5. Preguntas "Gestión de Pruebas"

### 5.0 Pregunta 29

OA-5.1.4

K1

¿Cuál de las siguientes afirmaciones describe mejor cómo se dividen las tareas entre el gestor de pruebas y el probador?

Juego de Respuestas:

- a) El jefe de pruebas planifica las actividades de prueba y elige los estándares a seguir, mientras que el probador elige las herramientas y controles a utilizar.
- b) El jefe de pruebas planifica, organiza y controla las actividades de prueba, mientras que el probador especifica y ejecuta las pruebas.**
- c) El jefe de pruebas planifica, supervisa y controla las actividades de prueba, mientras que el probador diseña las pruebas y decide sobre la aprobación del objeto de prueba.
- d) El jefe de pruebas planifica y organiza las pruebas, y especifica los casos de prueba, mientras que el probador prioriza y ejecuta las pruebas.

Justificación:

- a) FALSO - Seleccionar herramientas es una tarea del jefe de pruebas (Sección 5.1.2).
- b) VERDADERO - véase el apartado 5.1.2.**
- c) FALSO - Decidir sobre la aprobación del objeto de prueba es una tarea del jefe de pruebas (vea la Sección 5.1.2).
- d) FALSO - El jefe de pruebas no especifica los casos de prueba (Sección 5.1.2).

### 5.1 Pregunta 30

OA-5.5.3

K1

¿Cuál de las siguientes opciones puede clasificarse como un riesgo de producto?

Juego de Respuestas:

- a) Baja calidad de requisitos, diseño, código y pruebas.
- b) Problemas de carácter político y retrasos en áreas especialmente complejas del producto.
- c) Áreas propensas a errores, daño potencial para el usuario, características de producto pobres.**
- d) Problemas en la definición de los requisitos adecuados, potenciales áreas de fallo en el software o sistema.

Justificación:

- a) FALSO - Los requisitos de baja calidad son un riesgo del proyecto (Sección 5.5.1).
- b) FALSO - Todos los elementos de esta opción son riesgos de proyecto (Sección 5.5.1).
- c) VERDADERO - todos los elementos son riesgos de producto (Sección 5.5.2).**
- d) FALSO - Los problemas de requisitos son el riesgo del proyecto. (Sección 5.5.1).

<b>5.2 Pregunta 31</b>
------------------------

OA-5.2.9	xx
----------	----

¿Cuáles de los siguientes elementos son criterios de salida típicos de las pruebas?

Juego de Respuestas:

- a) Medición de la cobertura de las pruebas, medición de fiabilidad, coste de las pruebas, calendario, estado de la corrección de defectos y riesgos residuales.**
- b) Mediciones de cobertura de pruebas, mediciones de fiabilidad, grado de independencia del probador y completitud del producto.
- c) Mediciones de cobertura de pruebas, mediciones de fiabilidad, coste de las pruebas, disponibilidad del código susceptible de prueba, tiempo de comercialización y completitud del producto.
- d) Tiempo de comercialización, defectos residuales, cualificación del probador, grado de independencia del probador, mediciones de la cobertura de la prueba y coste de la prueba.

Justificación:

- a) VERDADERO - véase la sección 5.2.4 (todos los 5 puntos).**
- b) FALSO - El grado de independencia del probador no juega un papel en los criterios de salida (vea Sección 5.2.4).
- c) FALSO - "disponibilidad de código susceptible de prueba" es un criterio de entrada (Sección 5.2.3).
- d) FALSO - El grado de independencia del probador, así como la cualificación del probador, no juegan un papel en los criterios de salida (vea Sección 5.2.4).

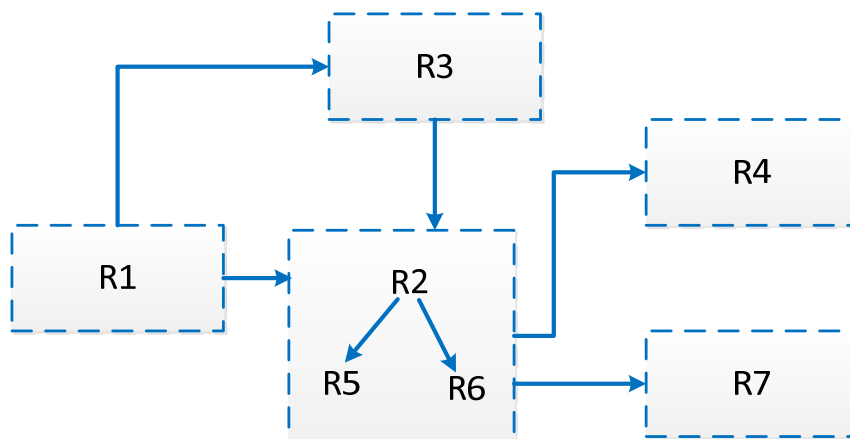
**5.3 Pregunta 32**

OA-5.2.5	K3
----------	----

Como Jefe de Pruebas, usted tiene los siguientes requisitos para probar:

- R1 - Anomalías de proceso
- R2 - Sincronización
- R3 - Confirmación
- R4 - Puntos problemáticos
- R5 - Datos financieros
- R6 - Datos de Diagrama
- R7 - Cambios en el perfil de usuario

La notación para indicar las dependencias lógicas de cualquier requisito es, por ejemplo, "R1 -> R3", lo que significa que R3 depende de R1.



¿Cuál de las siguientes opciones estructura el calendario de ejecución de pruebas de acuerdo con las dependencias de los requisitos?

Juego de Respuestas:

- a) R3 --> R2 --> R1 --> R7 --> R5 --> R6 --> R4
- b) R2 --> R5 --> R6 --> R4 --> R7 --> R1 --> R3
- c) **R1 --> R3 --> R2 --> R5 --> R6 --> R4 --> R7**
- d) R1 --> R2 --> R5 --> R6 --> R3 --> R4 --> R7

Justificación:

- a) FALSO - todo depende de R1, por lo que cualquier flujo de prueba que no se inicie en R1 es FALSO.
- b) FALSO - todo depende de R1, por lo que cualquier flujo de prueba que no se inicie en R1 es FALSO.
- c) **VERDADERO - las pruebas se especifican en una cadencia que tiene en cuenta las dependencias.**
- d) FALSO - R2 depende de R3, por lo que R3 debe probarse antes que R2.

<b>5.4 Pregunta 33</b>	
------------------------	--

OA-5.1.1	K1
----------	----

¿Cuál de las siguientes opciones es un posible beneficio de las pruebas independientes?

Juego de Respuestas:

- a) Se hace más trabajo porque los probadores no molestan a los desarrolladores todo el tiempo.
- b) **Los probadores independientes tienden a ser imparciales y a encontrar defectos diferentes que los desarrolladores.**
- c) Los probadores independientes no necesitan formación y preparación adicionales.
- d) Los probadores independientes reducen el cuello de botella en el proceso de gestión de incidencias.

Justificación:

- a) FALSO - independencia no significa la pérdida de cooperación.
- b) **VERDADERO - esa es una de las razones para la independencia (Sección 5.1.1).**
- c) FALSO - los probadores necesitan formación y preparación.
- d) FALSO - no hay conexión entre los probadores independientes y el cuello de botella en el proceso de gestión de incidencias.

<b>5.5 Pregunta 34</b>	
------------------------	--

OA-5.5.4	K1
----------	----

¿Cuál de las siguientes opciones es un riesgo de proyecto?

Juego de Respuestas:

- a) **Escasez de personal y competencias.**
- b) Características de software deficientes (por ejemplo, usabilidad).

- c) Software entregado con tendencia a fallos.
- d) Posible defecto de fiabilidad (bug).

Justificación:

- a. VERDADERO - La falta de competencias y de personal es un riesgo típico del proyecto (Sección 5.5.1).**
- b. FALSO - es un riesgo del producto (Sección 5.5.2).
- c. FALSO - es un riesgo del producto (Sección 5.5.2).
- d. FALSO - es un riesgo del producto (Sección 5.5.2).

### 5.6 Pregunta 35

OA-5.3.3

K2

Como jefe de pruebas, se le pide un informe de resumen de la prueba. En cuanto a las actividades de prueba, y de acuerdo con la norma IEEE 829, ¿cuál debería ser la información MÁS importante a incluir en su informe?

Juego de Respuestas:

- a) El número de casos de prueba ejecutados y sus resultados.
- b) Una visión general de las principales actividades de prueba, eventos y el estado con respecto al cumplimiento de los objetivos.**
- c) Evaluación global de cada elemento de trabajo de desarrollo.
- d) Formación impartida por los miembros del equipo de prueba para apoyar el esfuerzo de la prueba.

Justificación:

- a) FALSO - el número de casos de prueba ejecutados y sus resultados pueden incluirse en un informe de resumen de prueba según IEEE 829, pero no como la parte más importante.
- b) VERDADERO - el informe resumen de la prueba debe incluir información sobre las principales actividades, eventos y el estado con respecto al cumplimiento de los objetivos (véase también el Plan de Estudios, Sección 5.3.2).**
- c) FALSO - la evaluación de cada elemento de trabajo de desarrollo no forma parte del informe resumen de la prueba.
- d) FALSO - La formación no es relevante en un informe resumen de la prueba.

5.7 Pregunta 36

OA-5.6.2

K3

Usted es un probador en un proyecto de desarrollo de software de seguridad crítica. Durante la ejecución de una prueba, usted descubre que uno de sus casos de prueba falló, causando que usted escriba un informe de incidencia.

De acuerdo con el estándar IEEE Std. 829, ¿cuál debería considerar usted que es la información MÁS importante para incluir en su informe de incidencia en el contexto de un desarrollo crítico para la seguridad?

Juego de Respuestas:

- a) **Impacto, descripción de la incidencia, fecha y su nombre.**
- b) Identificador único para el informe, los requisitos especiales necesarios y la persona que causó el defecto.
- c) Elementos transmitidos, su nombre y sus sensación sobre la posible causa raíz del defecto.
- d) Descripción de la incidencia, entorno de desarrollo y resultados esperados de la prueba

Justificación:

- a) **VERDADERO - la información más importante que debe incluirse en un informe de incidentes para software crítico es el impacto.**
- b) FALSO - esta información debe estar en un reporte de incidencia, pero esta información no es tan importante como el impacto; vea a).
- c) FALSO - ver a), un informe de incidencia debe contener información objetiva, no el "sensación" del probador sobre la posible causa raíz.
- d) FALSE - esta información debe estar en un reporte de incidencia, pero esta información no es tan importante como el impacto; vea a).

## 6. Preguntas "Herramientas de Soporte de Pruebas"

### 6.0 Pregunta 37

OA-6.3.1

K1

De la siguiente lista, ¿cuáles son los principios recomendados para introducir una herramienta de prueba en una organización?

1. Desplegar la herramienta a toda la organización al mismo tiempo.
2. Comenzar con un proyecto piloto.
3. Adaptar y mejorar los procesos para adecuarlos al uso de la herramienta.
4. Proporcionar formación y entrenamiento para los nuevos usuarios.
5. Dejar que cada equipo decida su propia forma de utilizar la herramienta.
6. Supervisar que los costes no superen el coste de adquisición inicial.
7. Recopilar las lecciones aprendidas de todos los equipos.

Seleccionar UNA opción.

Juego de Respuestas:

- a) 1, 3, 4, 5
- b) 2, 5, 6
- c) **2, 3, 4, 7**
- d) 1, 6, 7

Justificación:

La afirmación 1 NO ES VERDADERA - Se recomienda hacer primero un despliegue piloto, antes de lanzarlo a toda la organización (Sección 6.3).

La afirmación 2 es VERDADERA - Ver arriba.

La afirmación 3 es VERDADERA - "Evaluar cómo la herramienta se ajusta a los procesos y prácticas existentes, y determinar qué necesitaría cambiar" (Sección 6.3).

La afirmación 4 es VERDADERA - La capacitación es uno de los factores de éxito para el despliegue (Sección 6.3).

La afirmación 5 NO ES VERDADERA - Si usted deja que todos decidan cómo usar la herramienta, habrá desorden. "Definir las normas de uso" es uno de los factores de éxito para el despliegue (Sección 6.3).

Afirmación 6 NO ES VERDADERA - El coste de desplegar una herramienta es más que simplemente el coste de adquisición de la herramienta. Fallar en darse cuenta de esto es uno de los riesgos asociados con el despliegue de herramientas (Sección 6.2).

La afirmación 7 es VERDADERA - "Recopilar las lecciones aprendidas de todos los equipos" es uno de los factores de éxito para el despliegue (Sección 6.3).

Por lo tanto

- a) FALSO
- b) FALSO
- c) **VERDADERO**
- d) FALSO

6.1 Pregunta 38
-----------------

OA-6.2.2	K1
----------	----

¿Cuál de las siguientes opciones describe MEJOR una característica de una herramienta de ejecución de pruebas basada en palabras clave?

Juego de Respuestas:

- a) **Una tabla con datos de entrada de prueba, palabras de acción y resultados esperados controla la ejecución del sistema sujeto a prueba.**
- b) Las acciones del probador se automatizan usando un script que se vuelve a ejecutar varias veces.
- c) Las acciones del probador se automatizan utilizando un guion que se ejecuta con varios conjuntos de datos de entrada de prueba.
- d) La capacidad de registrar los resultados de las pruebas y compararlos con los resultados esperados almacenados en un archivo de texto.

Justificación:

- a) **VERDADERO - "En un enfoque de prueba basado en palabras clave, la hoja de cálculo contiene palabras clave que describen las acciones a tomar (también llamadas palabras de acción) y datos de prueba" (Sección 6.2.2).**
- b) FALSO - esta es una descripción de la automatización de pruebas mediante guión (Sección 6.2.2.2).
- c) FALSO - esta es una descripción de la automatización de pruebas guiadas por datos (Sección 6.2.2.2).
- d) FALSO - esto describe una parte de lo que hace un marco de trabajo para automatización de pruebas (Sección 6.1.6).



## 6.2 Pregunta 39

OA-6.3.2

K1

¿Cuál de los siguientes NO es un objetivo de un proyecto piloto para la evaluación de una herramienta?

### Juego de Respuestas:

- a) Evaluar cómo se adapta la herramienta a los procesos y prácticas existentes.
- b) Determinar el uso, gestión, almacenamiento y mantenimiento de la herramienta y los productos de prueba.
- c) Evaluar si los beneficios se obtendrán a un coste razonable.
- d) Reducir la tasa de defectos en el proyecto piloto.

### Justificación:

- a) FALSO - la afirmación es verdadera, (sección 6.3).
- b) FALSO - la afirmación es verdadera, (sección 6.3).
- c) FALSO - la afirmación es verdadera, (sección 6.3).
- d) VERDADERO - la reducción del número de defectos no es el objetivo de un proyecto piloto (ver sección 6.3).

## 6.3 Pregunta 40

OA-6.1.3

K2

Una organización de desarrollo de software y pruebas desea alcanzar los objetivos de mejora de la eficiencia de pruebas que se enumeran a continuación.

¿Cuál de las siguientes opciones estaría mejor soportada por una herramienta de gestión de pruebas?

### Juego de Respuestas:

- a) **Permitir la trazabilidad entre requisitos, pruebas y defectos (bugs).**
- b) Optimizar la capacidad de las pruebas para identificar fallos.
- c) Resolver defectos más rápido.
- d) Automatizar una selección de casos de prueba para su ejecución.

Justificación:

- a) **VERDADERO** - porque la trazabilidad entre los requisitos y las pruebas es una funcionalidad de una herramienta de gestión de pruebas (sección 6.1.3).
- b) FALSO - porque esto no es posible con las herramientas de gestión de pruebas (sección 6.1.6).
- c) FALSO - porque esto no se resuelve, básicamente, con herramientas de gestión de pruebas (sección 6.1.3).
- d) FALSO - porque la selección de casos de prueba no está soportada por herramientas de gestión de pruebas (sección 6.1.6).