

Nombre:

Apellidos:

EXAMEN FINAL ORDINARIO

[Se valorará especialmente la capacidad de síntesis, con ideas claras, breves y bien estructuradas]

Pregunta 1ª (1'5 puntos)

Explicar en qué consisten las pruebas unitarias en sistemas orientados a objetos. Indicar sobre que tipo de artefactos software se aplican y de qué manera (caja blanca vs caja negra).

(utilizar únicamente el espacio anterior con letra clara y legible)

Nombre:

Apellidos:

[Se valorará especialmente la capacidad de síntesis, con ideas claras, breves y bien estructuradas]

Pregunta 2ª (1'5 puntos)

Explicar las ventajas de la técnica de Definición del Alcance basada en DFTs frente a los tradicionales WBSs.

(utilizar únicamente el espacio anterior con letra clara y legible)

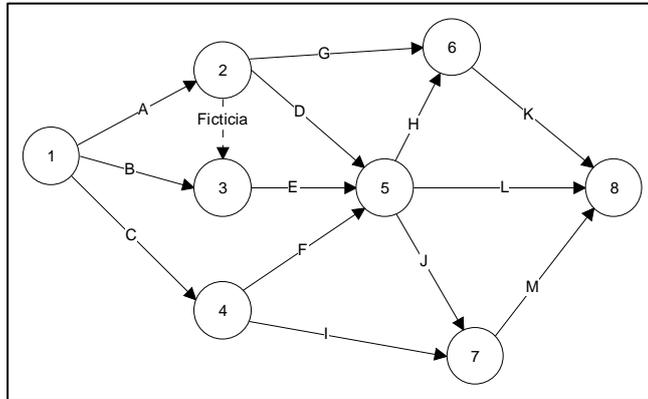
Nombre:

Apellidos:

[Se valorará especialmente la capacidad de síntesis, con ideas claras, breves y bien estructuradas]

Pregunta 3^a (1'5 puntos)

Dibujar el diagrama de precedencias equivalente al diagrama ADM de la figura. Explicar porqué en ambos casos solo puede haber un único inicio.



(utilizar únicamente el espacio anterior con letra clara y legible)

Nombre:

Apellidos:

Ejercicio 1º (5'5 puntos)

Se pretende construir la aplicación GPMS para gestionar el proceso de mantenimiento del software en una empresa que mantiene software de otras organizaciones. Para ello se ha mantenido una reunión con diversas personas de la empresa, resultado de la cual se ha elaborado un primer documento llamado "Requisitos de GPMS, versión 1" (se adjunta después).

Se pide estimar el tamaño del software en puntos función sin ajustar, ajustados y en líneas de código fuente.

NOTA: Cualquier consideración adicional que se deba realizar debe reflejarse explícitamente. Se valorará que se consideren circunstancias o valores realistas.

Requisitos de GPMS, versión 1

El proceso de mantenimiento definido es el siguiente:

- 1) Un usuario presenta una petición de modificación para que se modifique el software con el que trabaja.
- 2) Un "gestor de peticiones" recibe la petición y decide qué tipo de mantenimiento debe aplicarse. Se sabe que hay dos tipos posibles de mantenimiento: planificable y no planificable. Para cada uno de ellos se han definido una serie de actividades que deben ejecutarse. Las actividades del planificable son distintas del no planificable.
- 3) Una vez decidido el tipo de mantenimiento, el gestor de peticiones entrega la petición al equipo de mantenimiento. Éste inspecciona el software que debe mantener, implementa los cambios requeridos, realiza las pruebas necesarias y pasa el producto intervenido al entorno de producción. A continuación, registra la petición como "servida".

Se desea que la aplicación que se construya mantenga el control absoluto de todas las peticiones de modificación. Es decir, que la aplicación debe ser capaz de responder, como mínimo, a las siguientes preguntas:

- 1) Situación en que se encuentra una petición de modificación, indicando última actividad y tarea ejecutadas y fecha en que se inició la tarea actual.
- 2) Recursos y tiempo consumidos por una petición.
- 3) Recursos y tiempo consumidos por cierta tarea o actividad de cierta petición
- 4) Recursos y tiempo consumidos en un periodo
- 5) Recursos y tiempo consumidos por una aplicación o base de datos.
- 6) Número de peticiones sufridas por una aplicación.
- 7) Número total de peticiones recibidas.
- 8) Número total de peticiones realizadas por un usuario
- 9) Número de peticiones de cada tipo de mantenimiento.
- 10) Recursos y tiempo consumidos por las peticiones de cada tipo de mantenimiento.
- 11) Número de páginas de documentación generadas por cada petición.
- 12) Valores máximos, mínimos, medios, etc. de todos los puntos anteriores.

Además, debe ser capaz de generar listados para todos los puntos anteriores. Es muy importante la seguridad del sistema desde dos puntos de vista: concesión de permisos de acceso dependiendo del usuario de la aplicación y seguridad frente a posibles pérdidas de información.

El software se desarrollará en C++ en entorno Windows, con arquitectura cliente/servidor, utilizando protocolo ODBC para acceder al SGBD relacional. El equipo de desarrolladores tiene experiencia media.