

Módulo: ENTORNOS DE DESARROLLO

I.E.S.: AGUADULCE

C.F.G.S. : DESARROLLO DE APLICACIONES WEB

Fecha: 2 DE FEBRERO DE 2012

Hora de Comienzo: 9:00 H

Examen: Parte 1 (TEÓRICA)

Duración: 2 HORAS

Nombre:

D.N.I.:

Centro en el que se realiza el examen:

INSTRUCCIONES:

Marca el profesor que tengas asignado:

Fran []

José Luis []

La puntuación total del examen PARTE 1 + PARTE 2 será de 10 puntos. Parte teórica: 5 puntos. Parte Práctica: 5 puntos.

La nota del examen se calculará como la media ponderada de la parte teórica (50% del total) y la parte práctica (50% del total), siempre y cuando la nota de cada parte sea mayor o igual que 3.

Para el examen práctico se podrá hacer uso del material bibliográfico y digital que se estime oportuno, así como de apuntes. No obstante, se advierte del peligro de pérdida de tiempo que conlleva ponerse a consultarlo durante el examen, pudiendo consumirse el tiempo disponible en la consulta, y quedándose sin tiempo para las respuestas.

PARTE 1: CUESTIONES TEÓRICAS: (5 puntos) (Cada pregunta correcta puntuará con 0,1 puntos, cada incorrecta restará 0,05 puntos. Si de deja sin contestar ni suma ni resta)

1. ¿Qué lenguaje es directamente ejecutable por la computadora?

- a. Fuente.
- b. **Máquina.**
- c. Ensamblador.
- d. Objeto.

2. Tener deficiencias en la fase de _____ es la principal causa del gran porcentaje de fracasos de los proyectos software.

- a. Diseño
- b. Codificación
- c. Documentación
- d. **Análisis**

3. ¿Qué componente es vital para unir archivos en la generación del ejecutable?

- a. Compilador.
- b. **Linker.**
- c. Ensamblador.
- d. Intérprete.

4 La etapa consistente en dividir el problema general en partes con funciones definidas es:

- a. El análisis.
- b. **El diseño.**
- c. La programación.
- d. La ejecución.

5. El entorno de ejecución de aplicaciones está compuesto por _____ y la máquina virtual del lenguaje de programación.

- a. El enlazador
- b. El sistema operativo
- c. El lenguaje de programación
- d. **El API**

6. El documento de la guía técnica contiene información:

- a. Para que los usuarios de la aplicación sepan utilizarla.
- b. Para que los desarrolladores y usuarios finales instalen la aplicación.
- c. **Para que los analistas y desarrolladores pueden mantener la aplicación.**
- d. Para analistas y usuarios finales.



7. ¿Cómo se llama el cambio consistente en mejorar la funcionalidad de todo software?
- Correctivo.
 - Perfectivo.**
 - Adaptativo.
 - Evolutivo.
8. En la etapa de compilación el tipo de código que se obtiene es:
- Código Objeto.**
 - Código Fuente.
 - Código ejecutable.
 - Código ejecutable directo.
9. De todos los tipos de software, _____ es el encargado de gestionar los recursos hardware de un computador.
- La aplicación.
 - El compilador.
 - El código objeto.
 - El sistema operativo.**
10. En un supuesto práctico, ¿qué tipo de requisito es el deseo del cliente de incluir en la aplicación el control de stock de productos en el almacén?
- No funcional.
 - Funcional.**
 - No es un requisito.
 - Depende de la fase de codificación.
11. ¿Cómo se llama el proceso de traducción de código fuente a código objeto?
- Interpretación.
 - Compilación.**
 - Ejecución.
 - Runtime Environment.
12. ¿En qué momento se suelen realizar las pruebas Beta Test?
- Antes de las pruebas de integración.
 - En el mismo entorno productivo de la aplicación, una vez instalada.**
 - Después de realizar las pruebas unitarias.
 - lo largo de la etapa de mantenimiento.
13. La realización de pruebas _____ nos permite detectar errores de cada parte del programa por separado.
- semánticas
 - de integración
 - funcionales
 - unitarias**
14. El modelo en cascada con realimentación es
- El modelo perfecto si los requisitos están claros y el proyecto es rígido**
 - El modelo idóneo si se prevén muchos cambios durante el desarrollo.
 - Es el modelo de vida clásico del software.
 - Es un tipo de modelo evolutivo.
15. El modelo en espiral
- Se trata de varios ciclos en cascada que se repiten y se refinan en cada incremento.
 - Se divide en 6 zonas, llamadas regiones de tareas.**
 - Es un modelo bastante sencillo.
 - Es un modelo rígido que permite pocos cambios.



16. Una máquina virtual

- a. Es un tipo especial de Hardware que se instala en el ordenador para poder ejecutar los programas.
- b. **Es un tipo especial de software cuya misión es separar el funcionamiento del ordenador de los componentes hardware instalados.**
- c. Funciona como una capa de Hardware de bajo nivel y actúa como puente entre el bytecode de la aplicación y los dispositivos físicos del sistema.
- d. Es un tipo de software que se utiliza para compilar los programas de código fuente a código objeto.

17. Entre las ventajas de utilizar Frameworks tenemos:

- a. Poca dependencia del código.
- b. Consume pocos recursos.
- c. **Permite la reutilización de partes de código para otras aplicaciones.**
- d. Poco portable.

18. ¿Qué componentes de los IDE permite realizar la escritura del código?

- a. Depurador.
- b. **Editor de textos.**
- c. Compilador.
- d. Intérprete.

19. Tener previamente instalado _____ es imprescindible para poder instalar y ejecutar NetBeans.

- a. Linux
- b. JVM
- c. JRE
- d. **JDK**

20. ¿Qué componente es responsable del seguimiento de las variables en tiempo de ejecución?

- a. Compilador.
- b. **Depurador.**
- c. Editor de textos.
- d. Intérprete.

21. De las funcionalidades siguientes, cual no se pueden conseguir añadiendo plugins a nuestro entorno.

- a. **Reducción de tiempo de ejecución.**
- b. Posibilidad de importar proyectos de otros lenguajes de programación.
- c. Refactorización de programas.
- d. Utilidades para la realización de pruebas al software.

22. La extensión propia de los módulos y plugins en NetBeans es:

- a. .nmb.
- b. **.nbm.**
- c. ~.jar.
- d. ~tar.gaz.

23. ¿Cómo se llama el primer software que se consideró el precursor de los actuales IDE, en la década de los 70?

- a. MSDOS.
- b. **Maestro.**
- c. Cliente.
- d. Evolutivo.

24. El lenguaje _____ se considera que fue el primero en usar un IDE.

- a. C++
- b. JavaScript
- c. PHP
- d. **BASIC**



25. En Linux no podemos instalar el entorno:

- a. **Visual Studio.**
- b. NetBeans.
- c. Eclipse.
- d. Gambas.

26. ¿Cómo se llama el cambio consistente en mejorar la legibilidad de programas, sin alterar la funcionalidad del mismo?

- a. Correctivo.
- b. **Refactorización.**
- c. Adaptativo.
- d. Evolutivo.

27. La extensión _____ es característica de proyectos empaquetados en Java.

- a. .java
- b. .class
- c. .gaz
- d. **.jar**

28. De los siguientes lenguajes, cual no es orientado a objetos?

- a. **Lenguaje C.**
- b. Lenguaje C++.
- c. Lenguaje Java.
- d. Lenguaje PowerBuilder.

30. ¿Cómo se llama la adición de plugins sin salir del IDE?

- a. Off-line.
- b. **On-line.**
- c. Instantánea.
- d. Demorada.

31. ¿Cómo se llama el proceso de alteración de código para mejorar su legibilidad, sin cambiar su funcionalidad asociada?

- a. Depuración.
- b. **Refactorización.**
- c. Ejecución.
- d. Compilación.

32. ¿Qué es el llamado manifest file ?

- a. Un archivo de manifiesto.
- b. **Archivo especial que identifica a un módulo.**
- c. Archivo resultado de la compilación de un programa.
- d. Un archivo ejecutable.

33. De las siguientes funciones, cual de ellas no se puede realizar con el editor de textos en el IDE:

- a. **Conseguir la compilación de las aplicaciones.**
- b. Resaltar y colorear la sintaxis del código.
- c. Inserción automática de paréntesis y corchetes.
- d. Proponer sugerencias de resolución de problemas.

34. ¿En qué momento se realiza la ejecución de un programa?

- a. Antes de depurar el programa.
- b. **Una vez corregido, compilado y depurado el programa.**
- c. Antes de compilar el programa.
- d. Es indiferente el momento de la ejecución.



35. El archivo llamado _____ es el archivo principal de una aplicación Java.
- java.main
 - java.first
 - include
 - Main.java**
36. ¿Cual de los siguientes entornos de desarrollo es propietario?
- Netbeans.
 - Eclipse.
 - JCreator**
 - Gambas.
37. La prueba de software.
- Solo sirve para verificar el sistema.
 - Sirve para verificar y validar el sistema.**
 - Solo sirve para validar el sistema.
 - Su realización es opcional.
38. En la planificación de pruebas.
- Se depura el programa.
 - Se realiza la documentación de las pruebas.
 - Se establecen puntos de ruptura en el código.
 - Se diseñan los tipos de prueba y los casos de prueba.**
39. ¿Qué componente del IDE es básico en la realización de pruebas?
- Compilador.
 - Depurador.**
 - Ensamblador.
 - Intérprete.
40. La herramienta de prueba unitaria más extendida en Java es
- Es SimpleTest.
 - El JUnit.**
 - El NUnit.
 - MOQ.
41. La regresión es
- un tipo de prueba de validación.
 - un proceso que se realiza cuando se produce un cambio en el código.**
 - un paso necesario en la depuración del programa.
 - un proceso asociado a la fase de diseño del proyecto.
42. Un caso de prueba.
- Es cada fase de la planificación de pruebas.
 - Es cada elemento que interviene en la depuración.
 - Es diseñado con la colaboración del cliente.
 - Se diseña intentando que la probabilidad de detección de errores sea máxima.**
43. ¿Cómo se llama la prueba que comprueba el cumplimiento de los requisitos funcionales?
- Regresión.
 - Validación.**
 - Integración.
 - Sistema.



44. El objetivo del cubrimiento

- a. es obtener casos de prueba representativos.
- b. **comprobar que todos los caminos se pueden ejecutar.**
- c. establecer casos de prueba con valores en el límite del rango.
- d. establecer clases de equivalencia que disminuyan el número prueba.

45. Las clases de equivalencia.

- a. Nos ayudan a diseñar casos de prueba con valores límite.
- b. Nos permiten validar el sistema.
- c. Son herramientas de depuración.
- d. **Nos permite crear casos de prueba representativos de un conjunto de valores posibles.**

46. Con las clases de equivalencia diseñamos casos de pruebas

- a. con valores fuera del rango admitido.
- b. **con valores representativos del rango admitido.**
- c. con valores en el límite del rango admitido.
- d. todas las respuestas anteriores no son válidas.

47. ¿Qué herramienta de automatización de pruebas no es para Java?

- a. Unit.
- b. **FoxUnit.**
- c. TestNG.
- d. JTiger.

48. ¿En qué momento se suelen realizar las pruebas de la unidad?

- a. **Antes de las pruebas de integración.**
- b. En el mismo entorno productivo de la aplicación, una vez instalada.
- c. Después de realizar las pruebas unitarias.
- d. A lo largo de la etapa de mantenimiento.

49. La realización de pruebas _____ nos permite detectar errores de cada parte del programa por separado.

- a. semánticas.
- b. de integración.
- c. validación.
- d. **unitarias.**

50. Las pruebas funcionales.

- a. **son pruebas de caja negra.**
- b. son pruebas de caja blanca
- c. analizan y prueban directamente el código de la aplicación.
- d. Todas las respuestas anteriores son falsas.

