



Universidad de Valladolid

Departamento de Informática

Programación II, 1º I.T.I. de Sistemas

Convocatoria ordinaria.

7 de junio de 2004

Apellidos

Nombre

D.N.I.

- 1.- El tiempo disponible para la realización del examen es de 1¼ horas.
- 2.- Sólo hay una respuesta válida para cada pregunta tipo test.
- 3.- La suma de puntuaciones de las preguntas es de 8,5 puntos. Los puntos restantes (1,5) corresponden a la práctica. Las respuestas erróneas no restan puntos.
- 4.- Si en una pregunta tipo test desea cambiar la respuesta, marque la casilla válida e indique que es la correcta mediante una flecha con el texto "Si".
- 5.- Si presenta práctica, entréguela junto con el enunciado del examen.

1. (0.5 ptos) Indique cuál de los siguientes enunciados es falso:

- ☐ La etapa de análisis determina los requisitos de la aplicación.
- ☐ La metodología de *programación extrema* recomienda el diseño de la batería de pruebas de un módulo antes de su implementación.
- ☒ El diseño descendente obtiene una mejor reusabilidad que el diseño ascendente.
- ☐ El desarrollo más adecuado es aquel en que los factores de calidad se mantienen constantes y se va incrementando la funcionalidad.

2. (0.5 ptos) Indique cuál de los siguientes enunciados es verdadero:

- ☐ El documento de pruebas pertenece a la documentación del usuario.
- ☐ El documento de descripción funcional pertenece a la documentación del sistema.
- ☒ El documento de especificaciones pertenece a la documentación del sistema.
- ☐ Ninguno de los anteriores.

3. (0.5 ptos) Se dice que una aplicación es **robusta** cuando..

- ☐ No contiene fallos.
- ☐ Ha sido desarrollada mediante técnicas de *programación defensiva*..
- ☐ Ha sido desarrollada mediante técnicas de *programación bajo contrato*.
- ☒ Se comporta de manera adecuada en situaciones no contempladas en las especificaciones.

4. (0.4 ptos) El módulo A recibe un parámetro entero, n , con la precondition $\{n < 100\}$ y devuelve el valor n^2 . El módulo B llama al módulo A con un valor del parámetro de $n = 1000$ y obtiene como resultado el valor (correcto) 1000000. ¿Cuál es la interpretación **más adecuada** de este hecho según al paradigma de *programación bajo contrato*?

- ☐ El módulo A es erróneo, pues debería haber provocado un error en la aplicación al no cumplir B su precondition.
- ☐ El módulo B es erróneo, pues debería haber provocado un error en la aplicación al no cumplir la precondition de A.
- ☐ El módulo A actúa correctamente ya que el hecho de que B no cumpla su precondition no le exime de cumplir su postcondición.
- ☒ El módulo B es erróneo al no respetar la precondition de A. El módulo A puede devolver cualquier valor o fallar.

5. (0.7 ptos) En una aplicación de dibujo de diagramas existe un módulo, **nueva_figura**, que realiza las siguientes acciones:

- Insertar una nueva figura en la estructura de datos que representa el diagrama.
- Actualizar la variable que almacena el número de figuras del diagrama (éste valor lo obtiene de la acción anterior).
- Redibujar el diagrama en la pantalla.

Indique en el siguiente recuadro cuál sería el nivel de cohesión del módulo:

SECUENCIAL / COMUNICACIONAL / TEMPORAL

6. (0.5 ptos) Un programador debe elegir entre dos posibilidades a la hora de implementar un módulo que busca una palabra en un texto:

- a) Crear un módulo con dos parámetros, la palabra y el texto, el cuál devuelve la posición de la primera aparición de la palabra en el texto (o -1 si no aparece).
- b) Crear un módulo con tres parámetros: la palabra, el texto y un índice entero, i , que devuelva la posición de la primera aparición de la palabra en la parte del texto que va del carácter i -ésimo en adelante (o -1 si no aparece).

Este módulo será utilizado por otros dos: Uno que se encarga de detectar si existe o no una determinada palabra en un texto y otro que se encarga de buscar y reemplazar todas las apariciones de una palabra en un texto por otra palabra.

¿Cuál de las siguientes afirmaciones es verdadera?

- ☐ El módulo (a) tiene mayor cohesión que el módulo (b).
- ☐ El módulo (a) tiene menor cohesión que el módulo (b).
- ☒ La opción (a) da lugar a un menor acoplamiento que (b).
- ☐ La opción (a) da lugar al mismo acoplamiento que (b).

7. (0.5 ptos) Respecto al problema planteado en la pregunta anterior, indique cuál de los siguientes enunciados es verdadero:

- ☒ La opción (b) es más reusable.
- ☐ La opción (b) es más robusta.
- ☐ La opción (b) incrementa la facilidad de uso.
- ☐ Ninguno de los anteriores.

8. (0.5 ptos) Se dispone de dos especificaciones para un módulo que busca la posición de un valor en un vector:

- a) Precondición: Ninguna. Postcondición: Si el valor no existe en el vector, devuelve -1. Si existe, devuelve la primera posición del vector donde se encuentra un elemento igual al valor.
- b) Precondición: Vector ordenado. Postcondición: Si el valor no existe en el vector, devuelve -1. Si existe, devuelve la una posición cualquiera de entre aquellas que almacenan un elemento igual al valor.

Indique cual de los siguientes enunciados es verdadero:

- ☒ La especificación (a) es más fuerte que (b).
- ☐ La especificación (b) es más fuerte que (a).
- ☐ Ambas especificaciones son igual de fuertes.
- ☐ Las especificaciones son incompatibles.

9. (0.5 ptos) Indique cuál de los siguientes enunciados es verdadero:

- ☒ Una clase se denomina *abstracta* si no implementa alguno de sus métodos.
- ☐ Una clase es una instancia de un objeto.
- ☐ Además de las estructuras de control de la programación estructurada, la programación orientada a objetos define estructuras de control adicionales.
- ☐ El *downcasting* es un mecanismo que permite hacer que un objeto de clase A se comporte como un objeto de otra clase cualquiera.

10. (0.5 ptos) Se dispone de una clase A que implementa un método, *identidad*, que es una función que devuelve el carácter "A". La clase B hereda de A y redefine *identidad* para que devuelva el carácter "B". ¿Cuál sería el resultado del siguiente código en Eiffel? **Nota:** Se supone que *make* es un método de creación tanto de A como de B.

```
...
local x : A; y : B;
do
  create y.make;
  io.put_character(y.identidad);
  x := y;
  io.put_character(x.identidad);
...
```

- ☐ AB
- ☐ BA
- ☐ AA
- ☒ BB

11. (0.5 ptos) Una clase es genérica si...

- ☐ Las variables cuyo tipo es esa clase almacenan los valores de los objetos en lugar de referencias a los objetos.
- ☐ Es la raíz de la jerarquía de herencia.
- ☒ Está parametrizada por otras clases que se proporcionan en la declaración de variables cuyo tipo sea esa clase.
- ☐ No redefine los métodos heredados.

Firma del Alumno: (0.1 ptos)

12. (0.5 ptos) Indique cuál de las siguientes afirmaciones es falsa:

- ☒ Una clase expandida no puede definir métodos, sólo atributos.
- ☐ Una variable de una clase expandida almacena automáticamente un objeto de esa clase, sin necesidad de crearlo.
- ☐ Las variables cuyo tipo es una clase expandida almacenan valores de objetos en lugar de referencias.
- ☐ Las clases expandidas hacen un uso más eficiente de la memoria.

13. (0.5 ptos) Se define la clase A de la siguiente forma:

```
class A[B -> C]
inherits D
...
end -- A
```

¿Que condición debe cumplirse para que la siguiente declaración sea correcta?

x : A[E];

- ☐ E herede de A.
- ☐ E herede de B.
- ☒ E herede de C.
- ☐ E herede de D.

14. (0.5 ptos) El objetivo de una batería de pruebas es...

- ☒ Encontrar la mayor cantidad posible de fallos.
- ☐ Certificar que el módulo o aplicación está libre de fallos.
- ☐ Corregir los fallos del módulo o aplicación.
- ☐ Poner en contacto al usuario con la aplicación para que detecte los fallos existentes.

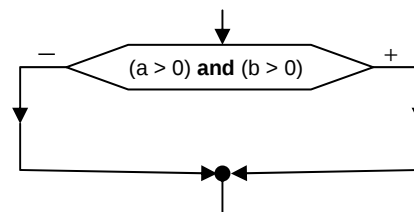
15. (0.5 ptos) La función **signo** recibe como entrada un valor entero, x, y devuelve el valor 1 si $x > 0$, el valor 0 si $x = 0$ y el valor -1 en caso contrario.

Indicar cuál de los siguientes valores de x podrían corresponder a los casos de prueba de una batería obtenida por el método de caja negra con análisis de valores límite:

- ☐ -1, 0, 1
- ☐ -10, -1, 1, 10
- ☐ -10, -5, -1, 0, 1, 5, 10
- ☒ -10, -1, 0, 1, 10

16. (0.8 ptos) ¿Cobertura de caminos implica cobertura de condiciones?

Si es así, escriba en el espacio restante una demostración abreviada. En caso contrario, indique un contraejemplo mediante un diagrama de flujo y una tabla de diseño de caja blanca en la que aparezcan los casos de prueba que consiguen cobertura de caminos pero no cobertura de condiciones.



a	b	C _{1a}	C _{1b}	Camino
1	1	T	T	+
1	-1	T	F	-