ESTRUCTURA DE DATOS Y ALGORITMOS **7-Febrero-2006**
Titulación: Ingeniero Técnico en Informática de Gestión **Curso: 2º**

Nombre-Apellidos:	**Nota:**

1. Ejercicio (1 punto)

Responder a las siguientes preguntas de tipo test, seleccionando sólo una de las respuestas. Se evaluarán sólo las respuestas contestadas con el siguiente criterio: respuesta acertada **positivo**, respuesta fallada **negativo**.

1. Un vector tiene la siguiente propiedad:
    a. Inserciones y borrados eficientes en medio de la estructura
    b. Inserciones y borrados eficientes en ambos extremos de la estructura
    c. Inserciones y borrados eficientes al final de la estructura
    d. Ninguna de las anteriores

2. Un vector se puede recorrer de la siguiente forma:
    a. Secuencial
    b. Bidireccional
    c. Saltos aleatorios
    d. Todas las anteriores

3. Una lista tiene la siguiente propiedad:
    a. Inserciones y borrados eficientes en medio de la estructura
    b. Inserciones y borrados eficientes al principio de la estructura
    c. Inserciones y borrados eficientes al final de la estructura
    d. Todas las anteriores

4. Una lista enlazada simple tiene la siguiente propiedad:
    a. Acceso secuencial
    b. Acceso bidireccional
    c. Acceso aleatorio
    d. Todas las anteriores

5. La probabilidad de no encontrar suficiente memoria disponible
    a. Es mayor para un vector que para una lista
    b. Es mayor para una lista que para un vector
    c. La probabilidad es la misma para el vector que para la lista
    d. Si la lista es doblemente enlazada B es cierto; si es simplemente enlazada A es cierto

6. Las pilas son conocidas porque realizan las operaciones de inserción y eliminación de acuerdo al principio de:
    a. FIFO
    b. FILO
    c. LIFO
    d. LILO

7. Las colas son conocidas porque realizan las operaciones de inserción y eliminación de acuerdo al principio de:
    a. FIFO
    b. FILO
    c. LIFO
    d. LILO

8. Las tablas hash son utilizadas porque:
    a. Se deben de recorrer los elementos en orden
    b. No se conoce el número de elementos a priori
    c. Se quiere utilizar un mismo índice para varios elementos
    d. Ninguna de las anteriores

9. Cual de las siguientes afirmaciones es cierta:
    a. Las estructuras de datos almacenan operaciones y las aplican sobre un objeto
    b. Las estructuras de datos restringen el uso de las variables
    c. Las estructuras de datos almacenan variables de objetos
    d. Las estructuras de datos almacenan referencias a objetos

10. Las principales diferencias entre las distintas estructuras son:
    a. La forma de organizar las operaciones y las restricciones que imponen sobre los datos
    b. La forma de organizar los datos y las restricciones que imponen sobre las operaciones
    c. La forma en que aplican la abstracción
    d. En principio, no existe ninguna diferencia sustancial, depende del problema planteado

ESTRUCTURA DE DATOS Y ALGORITMOS **7-Febrero-2006**
Titulación: Ingeniero Técnico en Informática de Gestión **Curso: 2º**

Nombre-Apellidos:	**Nota:**

1. Ejercicio (1 punto)

Responder a las siguientes preguntas de tipo test, seleccionando sólo una de las respuestas. Se evaluarán sólo las respuestas contestadas con el siguiente criterio: respuesta acertada **positivo**, respuesta fallada **negativo**.

11. Un vector tiene la siguiente propiedad:
    a. Ninguna de las siguientes
    b. Inserciones y borrados eficientes al final de la estructura
    c. Inserciones y borrados eficientes en ambos extremos de la estructura
    d. Inserciones y borrados eficientes en medio de la estructura

12. Un vector se puede recorrer de la siguiente forma:
    a. Todas las siguientes
    b. Saltos aleatorios
    c. Bidireccional
    d. Secuencial

13. Una lista tiene la siguiente propiedad:
    a. Todas las siguientes
    b. Inserciones y borrados eficientes en medio de la estructura
    c. Inserciones y borrados eficientes al principio de la estructura
    d. Inserciones y borrados eficientes al final de la estructura

14. Una lista enlazada simple tiene la siguiente propiedad:
    a. Todas las siguientes
    b. Acceso bidireccional
    c. Acceso secuencial
    d. Acceso aleatorio

15. La probabilidad de no encontrar suficiente memoria disponible
    a. Es mayor para una lista que para un vector
    b. Es mayor para un vector que para una lista
    c. La probabilidad es la misma para el vector que para la lista
    d. Si la lista es doblemente enlazada B es cierto; si es simplemente enlazada A es cierto

16. Las pilas son conocidas porque realizan las operaciones de inserción y eliminación de acuerdo al principio de:
    a. LIFO
    b. LILO
    c. FIFO
    d. FILO

17. Las colas son conocidas porque realizan las operaciones de inserción y eliminación de acuerdo al principio de:
    a. LIFO
    b. LILO
    c. FIFO
    d. FILO

18. Las tablas hash son utilizadas porque:
    a. Ninguna de las siguientes
    b. Se deben de recorrer los elementos en orden
    c. No se conoce el número de elementos a priori
    d. Se quiere utilizar un mismo índice para varios elementos

19. Cual de las siguientes afirmaciones es cierta:
    a. Las estructuras de datos almacenan operaciones y las aplican sobre un objeto
    b. Las estructuras de datos restringen el uso de las variables
    c. Las estructuras de datos almacenan referencias a objetos
    d. Las estructuras de datos almacenan variables de objetos

20. Las principales diferencias entre las distintas estructuras son:
    a. La forma de organizar las operaciones y las restricciones que imponen sobre los datos
    b. En principio, no existe ninguna diferencia sustancial, depende del problema planteado
    c. La forma en que aplican la abstracción
    d. La forma de organizar los datos y las restricciones que imponen sobre las operaciones

2. Ejercicio (3 puntos)

Una empresa de reparto de propaganda contrata a sus trabajadores por días. Cada repartidor puede trabajar varios días continuados o alternos. Los datos de los repartidores se almacenan en una lista enlazada simple. El programa a desarrollar contempla realizar las siguientes funcionalidades:

a) Crear una estructura de cola para recoger en ella el número de la seguridad social de cada repartidor y la entidad anunciada en la propaganda para un único día de trabajo. Pon un ejemplo para 4 propagandas.
b) Actualizar la lista citada anteriormente (que ya existe con contenido) a partir de los datos de la cola.

La información de la lista es la siguiente: número de seguridad social, nombre y total de días trabajados. Además, está ordenada por el número de la seguridad social. Si el trabajador no está incluido en la lista debe añadirse a la misma de tal manera que siga ordenada.

Nota: Implementa las estructuras de las clases necesarias para implementar las listas. Puedes utilizar los métodos de las listas y colas vistas en clase. Cualquier otra clase o método auxiliar debes de implementarlo.

3. Ejercicio. (2,5 puntos)

En la mayor parte de los lenguajes de programación se deben balancear los paréntesis (), [] y {}, y deben anidarse de forma correcta. Se define una cadena **correcta en paréntesis** como aquella que coincide con una de las siguientes pautas, en las que S representa una cadena (posiblemente vacía) sin símbolos de paréntesis, y P, P' y P'' representan, recursivamente, cadenas correctas en paréntesis.

- S
- P'(P)P''
- P'[P]P''
- P'{P}P''

Diseñe un algoritmo y escribe un método, que revise si una cadena es correcta en paréntesis, en un tiempo proporcional al tamaño de la cadena, usando una estructura vista en clase como estructura auxiliar de datos. Razona por qué has seleccionado la estructura de datos utilizada.

4. Ejercicio (3,5 puntos)

a) ***Diseñar e implementar en Java*** un método que escriba por salida estándar los términos de un polinomio **ordenados de forma decreciente** por su grado (valor del exponente). El polinomio es representado mediante una tabla hash de N=2*T posiciones, donde T es el número medio de términos en el polinomio.

public static void imprimir(Hashtable polinomio);

El tipo de los elementos en la tabla hash será:

```
class Termino {
      int exp;
      int coef;
}
```

Ejemplo:

Si el polinomio que nos han pasado como parámetro está representado con la siguiente tabla hash (para N= 10):

Indice	Clave	Datos
1	450	16
2	550	24
3	vacía	
4	100	- 20
5	0	8
6	vacia	
7	200	- 48
8	vacía	
9	vacía	
10	vacía	

El resultado que se mostrará en pantalla será el siguiente: $\mathbf{24x^{550} + 16\ x^{450} - 48x^{200} - 20x^{100} + 8}$

b) **Indicar de forma razonada**, suponiendo que todas las operaciones del TAD tablahash son de *O*(1), el **orden** de la operación realizada en el apartado anterior.

c) Explica brevemente por qué en la tabla hash anterior la información no aparece contigua en las casillas correspondientes y no está ordenada.

Nota: cualquier método auxiliar que sea utilizado, es necesario implementarlo

ANEXO I: Especificación de clases

```
class String {
String substring (int begin, int last);
//pos:Devuelve la secuencia de caracteres que se encuentra entre las posiciones begin y last,
//    las dos inclusive.
int compareTo (String s)
//pos:Devuelve 0 si el objeto actual es igual a s, -1 si es menor que s y 1 si es mayor que s.
```

```
class HashTable {
void insert (Object key, Object value);
//pos: inserta el elemento de clave key y datos value. Si la clave ya estaba en la tabla no --
//     hace nada.
void modify (Object key, Object value)
//pos: reemplaza los datos asociados a la clave key de la tabla por value. Si la clave no está
//     en la tabla devuelve no hace nada
Object find (Object key);
//pos: Devuelve los datos asociados a la clave key. Si la clave no está en la tabla devuelve
//     null.
void remove (Object key);
//pos: Borra de la tabla hash el elemento  que coincide con el valor de la clave key. Si la
//     clave no está en la tabla no hace nada.
Boolean exist (Object key);
//pos: Devuelve cierto si la clave key está en la tabla y falso en caso contrario
Enumeration keys ();
///pos: Devuelve un objeto para iterar sobre las claves de la tabla
Enumeration values(); //pos: Devuelve un objeto para iterar sobre los valores de la tabla
}
```