



Tiempo: 1 hora

Se recomienda leer atentamente los enunciados antes de contestar. No se permiten ni libros ni apuntes

La puntuación para preguntas con múltiples opciones es:

- Pregunta correcta: 1 punto
- Pregunta incorrecta: -0,25 puntos
- Pregunta en blanco: 0 puntos

Las preguntas en las que se marquen varias opciones serán consideradas incorrectas.

El examen se responderá en la misma hoja en el espacio reservado para ello.

Cualquier respuesta fuera de estas hojas será ignorada.

A

1.- Respecto de los Tipos Abstractos de Datos:

- a.- La utilización de los TAD aumenta, en general, la eficiencia de los programas.
- b.- Con la utilización de los TAD la programación se simplifica en general, aunque puede disminuir la eficiencia de los programas.
- c.- La utilización de los TAD, en general, dificulta la implementación de los programas.
- d.- La utilización de los TAD sólo es recomendable en el caso de la utilización de Pilas, Colas, Árboles y Grafos.

2.- ¿Cuál es el coste temporal de la operación ‘obtener valor mínimo’, en el caso medio, en las siguientes estructuras con ‘n’ elementos:

- | | | | | | |
|--------------------------------|---------|-------|----------|--------------|---------------------|
| a.- Pila: | O (cte) | O (n) | O (lg n) | O (n * lg n) | O (n ²) |
| b.- Cola: | O (cte) | O (n) | O (lg n) | O (n * lg n) | O (n ²) |
| c.- Árbol binario de búsqueda: | O (cte) | O (n) | O (lg n) | O (n * lg n) | O (n ²) |
| d.- Montículo de máximos: | O (cte) | O (n) | O (lg n) | O (n * lg n) | O (n ²) |
| e.- Montículo de mínimos: | O (cte) | O (n) | O (lg n) | O (n * lg n) | O (n ²) |

3.- Sea ‘f’ una función que se aplica sobre pilas. Di que tarea realiza

f (IniciarPila) -> 0

f (p) = CimaPila (p) + f (Desapilar (p))

4.- Supongamos un árbol binario de búsqueda en el que hemos insertado, y en ese orden, los siguientes valores enteros: 10, 2, 8, 5, 10, 9, 1, 13, 5.

¿Cuál o cuales de los siguientes recorridos será el recorrido prefijo del árbol?

- a.- 10, 2, 1, 8, 5, 5, 9, 10, 13
- b.- 1, 5, 9, 8, 2, 13, 10
- c.- 10, 2, 1, 8, 5, 9, 13
- d.- 10, 2, 8, 1, 5, 9, 13

¿Cuál será su recorrido infijo?

5.- Para realizar una función que elimine elementos repetidos de una lista con punto de interés...

- a.- ...Las operaciones básicas definidas en el TAD no son suficientes para implementarla. Se deberá implementar como operación básica del TAD.
- b.- ...El hecho de implementarla como operación básica del TAD, aumenta el coste espacial de la lista, ya que habría que añadir un segundo punto de interés a la lista.
- c.- ...Las operaciones básicas definidas en el TAD son suficientes para implementarla.
- d.- ...En ningún caso se puede implementar como un método de la clase.

6.- ¿Es posible representar/implementar la clase cola mediante listas de adyacencia?

- a.- No, ya que sólo es posible representar grafos mediante listas de adyacencia.
- b.- No, ya que sólo es posible representar estructuras no lineales, como árboles o grafos, mediante listas de adyacencia.
- c.- Sí, pero algunos métodos de la clase cola sería imposible implementarlos.
- d.- Sí, pero los métodos de encolar y desencolar podrían complicarse.



7.- Sea la siguiente declaración de tipos
Siga la següent declaració de tipus

```
typedef string Vec[10];
struct Reg {
    int num;
    Vec dato;
};
typedef Reg Vreg[10];
```

Si x es una variable de tipo Vreg, indicar el tipo resultante de las siguientes expresiones o si son incorrectas y por qué:
Si x és una variable de tipus Vreg, indicar el tipus resultant de les següents expressions o si son incorrectes i per què:

```
x[8] _____
* x[2].dato _____
x.dato[5] _____
x->dato _____
x[0].num _____
x[3].dato[5][2] _____
```

8.- Dado el siguiente programa:
Dona't el següent programa:

```
void f(int & x, int y);

int main(void)
{
    int x = 1; int y = 2;

    f(y, x);
    y = y + x;
    cout << y;
    return 0;
}

void f(int & x, int y)
{
    x = x + 3;
    y = y + 1;
}
```

¿Qué visualizará? Elegir una opción:
Què visualitzarà? Elegir una opció:

a) 3	b) 6	c) 7	d) 8
------	------	------	------

9.- Dado el siguiente vector:
Dona't el següent vector:

```
typedef char Vchar[7];
struct Reg
{
    Vchar v;
    float f;
}

typedef Reg Vvoro[10];
Vvoro dato;
```

Y sabiendo que el tamaño de un real son 6 bytes, el tamaño de un carácter es 1 byte y que la variable dato comienza en la posición de memoria 1000, calcular en que posición de memoria está $dato[3].v[5]$.

I sabent que la grandària d'un real son 6 bytes, la grandària d'un caràcter és 1 byte i que la variable dato comença en la posició de memòria 1000, calcular en quina posició de memòria està $dato[3].v[5]$.



Tiempo: 1'5 horas **Se recomienda leer atentamente los enunciados antes de contestar.** **No se permiten ni libros ni apuntes**

P1.- Realizar una función en C++ que devuelva el valor contenido en el nodo de grado uno de mayor nivel del árbol pasado como parámetro. La función devolverá 'error' si el árbol pasado estaba vacío o no existe ningún nodo de grado uno.

El prototipo de la función será:

```
bool ValorNodo (Arbol, Valor &);
```

Escribe el prototipo de los métodos de árboles que vayas a utilizar, así como el código de aquellas funciones auxiliares que creas conveniente.

P2.- Una cola con prioridades es una cola en la que los elementos no se insertan en el orden en que van llegando, sino que se insertan según un valor de prioridad asignado a cada elemento, insertándose los que tienen mayor prioridad más adelante en la cola que los que tienen menor prioridad. Si la prioridad es la misma, el nuevo elemento se inserta al final de la serie de elementos con su misma prioridad.

Realizar un nuevo método en C++, para la clase Cola que implemente este algoritmo de inserción con prioridad.

El prototipo del método será:

```
bool Cola::EncolarConPrioridad (Valor x)
```

siendo el tipo Valor:

```
struct Valor
{
    int prioridad;
    ValorInfo info;
};
```

Realizar dos versiones: Una para la representación estática de la cola y otra para la versión dinámica.

Escribe en ambos casos la parte privada y las definiciones de tipo necesarias, así como los prototipos de los métodos 'estandar' que vayas a utilizar y el código de los métodos auxiliares que creas conveniente.



P3.- Sea el siguiente programa en C++:

```
#include<iostream.h>
#include<string>

string f(string s)
{
    char c;
    string res;

    if(s == "")
        res = "";
    else
    {
        c = s[0];
        s.erase(0,1);
        res = f(s);
        res = res + c;
    }
    return res;
}

int main(void)
{
    string s = "hola";

    cout << f(s);
    return 0;
}
```

Realiza una traza del programa y di qué mostrará por pantalla.

erase(pos, num): Método que elimina **num** caracteres de un **string** empezando en la posición **pos** (inclusive).