

1. TAD

- Defina el concepto de Tipo Abstracto de Dato. Detalle las principales ventajas.
- Especifique e implemente el TAD Agenda de cumpleaños 2010. Esta agenda permite guardar los cumpleaños por día y mes. En la agenda solo se guarda el nombre de la persona que cumple los años. Dé una representación adecuada e implemente las operaciones de crear la agenda, agregar un cumpleaños en un determinado día y mes y determinar cuántas personas cumplen años en un determinado mes. Suponga que todos los meses tienen 30 días.

2. Tipos de datos

- Defina el concepto de tipo de datos.
- Distinga entre los tipos de datos simples y compuestos. Realice lo mismo entre los estándares y los definidos por el usuario. En cada caso ejemplifique.
- ¿Por qué le parece que son importantes los tipos de datos en programación?

3. Eficiencia

- ¿Cuándo podemos afirmar que un programa es eficiente?
- ¿Qué tipos de análisis se pueden realizar para determinar la eficiencia de un programa?
- Con respecto al ejer.1) si supone que los meses tienen diferente cantidad de días, proponga una representación posible. Reimplemente la operación de determinar cuántas personas cumplen años en un determinado mes y analice la eficiencia en cuanto a la memoria de ambas representaciones.

4. Listas

- Defina el concepto de listas simplemente enlazadas. Proponga una representación posible.
- Si se piensa en la operación de Agregar al final de la estructura, proponga una representación más eficiente en cuanto al tiempo y justifique.
- ¿Podría representar una lista simplemente enlazada utilizando un vector? Justifique.

5. Módulos

- ¿Qué es un módulo?
- ¿Por qué Ud. agregaría un módulo en la implementación de un programa? Ejemplifique
- ¿Cómo se comunican los módulos?

6. Recursión

¿Cuáles de los siguientes ejemplos son soluciones recursivas? ¿Por qué? Justifique a partir de la definición de recursión. ¿En caso que alguna no sea una solución recursiva, podría modificarse para que lo fuera, de qué forma?

```
i. Program ejemplo1;
Const max=10;
Type rango=0..max;
vector= record
    Datos: array [1..max] of char;
    DL: rango;
End;
Var pos: rango; v: vector;
Procedure Imprimir (v: vector; var pos: rango);
Begin
    Writeln( v.datos[pos]);
    pos:=pos+1;
    Imprimir(v,pos);
End;
Begin
    CargarVector(v);
    pos:=1;
    Imprimir(v,pos);
End.
```

```
ii. Program Ejemplo2;
Procedure Digitos( num:integer);
Begin
    If (num > 0) then
        Begin
            writeln(num mod 10);
            Digitos(num);
        End;
    End;
Begin
    Digitos(19876);
End.
```