



Cuadernillo de examen

ASIGNATURA	Fundamentos de Programación I	CÓDIGO	106
CONVOCATORIA	Parcial de Diciembre de 2003	PLAN DE ESTUDIOS	2002
ESPECIALIDAD	Común	CURSO	1º
TURNO	Mañana	CURSO ACADÉMICO	2003/2004
CARÁCTER	Cuatrimestral	PROGRAMA	Ingeniería Superior en Informática
DURACIÓN APROXIMADA	2 horas		

Soluciones propuestas al examen

Preguntas teórico-prácticas

Estructuras de control selectivas. Explique brevemente cada uno de los tipos de estructuras selectivas, su formato en diagrama de flujo, diagrama N-S y pseudocódigo y la utilidad de cada una de ellas.

Capítulo 5 del libro de texto y apuntes de clase

Aplicación

Se desea realizar un algoritmo que permita obtener el día de la semana a partir de un número entero introducido por teclado (0 para el domingo, 1 para el lunes, 2 para el martes, etc.). Diseñe dos versiones del algoritmo, una utilizando la estructura selectiva doble y otra utilizando la estructura selectiva múltiple.

```
//Versión con estructuras selectivas dobles
algoritmo DíaDeLaSemana1
var
    entero : día
inicio
    leer(día)
    si día = 0 entonces
        escribir('Domingo')
    si_no
        si día = 1 entonces
            escribir('Lunes')
        si_no
            si día = 2 entonces
                escribir('Martes')
            si_no
                si día = 3 entonces
                    escribir('Miércoles')
                si_no
                    si día = 4 entonces
                        escribir('Jueves')
                    si_no
                        si día = 5 entonces
                            escribir('Viernes')
                        si_no
                            escribir('Sábado')
                    fin_si
                fin_si
            fin_si
        fin_si
    fin
```

```
//Versión con estructura selectiva múltiple
algoritmo DíaDeLaSemana2
var
    entero : día
inicio
    leer(día)
    según_sea día hacer
```



```

0 : escribir ('Domingo')
1 : escribir ('Lunes')
2 : escribir ('Martes')
3 : escribir ('Miércoles')
4 : escribir ('Jueves')
5 : escribir ('Viernes')
si_no
  escribir ('Sábado')
fin_según
fin
  
```

Puntuación: 1,5 puntos

Preguntas prácticas

Una empresa quiere almacenar en una tabla las ventas realizadas por sus vendedores a lo largo de la semana. La empresa cuenta con 5 vendedores y en cada elemento i,j de la tabla se almacenará las ventas realizadas por el vendedor i el día j (j valdrá 1 para el lunes, 2 para el martes, 3 para el miércoles, etc.). Los nombres de los vendedores y el total de ventas semanales se guardarán en un array de registros.

	Nombre vendedor	Total ventas semanales							
1	...		2342	3212	4543	1543	3425	5453	1232
2	...		3423	5434	4534	3454	4343	6787	4343
3	...		3432	4323	6565	7878	9899	6554	3234
4	...		5434	3432	4322	3432	3423	5445	3432
5	...		5454	4345	4344	3234	5455	5454	5445

Se pide:

- Realizar las declaraciones de las estructuras de datos necesarias para la aplicación.
- Codifique un módulo que permita introducir los nombres de los vendedores y las ventas realizadas por cada uno de ellos y almacenarlos en las estructuras de datos.
- Codifique un módulo que obtenga el total de ventas mensuales de cada vendedor.
- Codifique un módulo que obtenga en nombre del vendedor que más ventas ha realizado en un día de la semana.

Puntuación: 3,5 puntos

Declaraciones de las estructuras de datos

```

tipos
  registro : vendedor
  cadena : nombre
  real : total
fin_registro
array[1..5] de vendedor : vendedores
array[1..5,1..7] de real : ventas
  
```

Introducir los nombres de los vendedores y las ventas

```

procedimiento LeerDatos(S vendedores : v; S ventas : t ; E entero m,n)
var
  entero : i, j
inicio
  desde i ← 1 hasta m hacer
    leer(v[i].nombre)
    desde j ← 1 hasta n hacer
      leer(t[i,j])
    fin_desde
  fin_desde
fin_procedimiento
  
```



Calcular el total de ventas semanales por vendedor

```
procedimiento CalcularVentasSemanales(S vendedores : v; S ventas : t ; E entero m,n)
var
  entero : i, j
inicio
  desde i ← 1 hasta m hacer
    v[i].total ← 0
    desde j ← 1 hasta n hacer
      v[i].total ← v[i].total + t[i,j]
    fin_desde
  fin_desde
fin_procedimiento
```

Obtener el nombre del vendedor con más ventas en un día

```
cadena función NombreVendedorMasVentas(S vendedores : v; S ventas : t ; E entero m,n)
var
  entero : i, j, FMax, CMax //FMax y CMax guardan la fila y columna de mayor valor
inicio
  FMax ← 1
  CMax ← 1
  desde i ← 1 hasta m hacer
    desde j ← 1 hasta n hacer
      si t[i,j] > t[FMax,CMax] entonces
        FMax ← i
        CMax ← j
      fin_si
    fin_desde
  fin_desde
  devolver(v[FMax].nombre)
fin_función
```