

INFORMATICA.

EXAMEN Septiembre 2003.

Pascal

Apellidos, Nombre

Grupo

1) (3.5 puntos) El conjunto de los números de *Hamming* (H) es el menor subconjunto de los números naturales verificando los siguientes axiomas:

- a) $1 \in H$
- b) Si $x \in H \Rightarrow 2x \in H$
- c) Si $x \in H \Rightarrow 3x \in H$
- d) Si $x \in H \Rightarrow 5x \in H$

Es decir, $H = \{1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, \dots\}$.

Escribe un programa en *Pascal* que, **utilizando el tipo SET**, genere un conjunto con los números de *Hamming* menores o iguales a 100. Una vez generado el conjunto, el programa deberá mostrar los elementos de éste por pantalla.

2) (4.5 puntos) Escribe una función mezclar, que tome como parámetros dos arrays con 10 elementos reales ordenados ascendentemente y devuelva como resultado el array de tamaño 20 con la mezcla ordenada de los elementos de los dos parámetros.

3) (2 puntos) Disponemos de dos ficheros binarios que contienen números naturales. Escribe un programa que genere un tercer fichero binario con todos los elementos del segundo fichero que superen la media de los elementos de primero.

INFORMATICA.

EXAMEN Septiembre 2003.

Haskell

Apellidos, Nombre

Grupo

1) (3 puntos) Sea el siguiente tipo para representar números racionales

```
infix 9 :/  
data Racional = Integer :/ Integer
```

donde el primer entero es el numerador y el segundo es el denominador. Haz este tipo instancia de las clases *Eq*, *Ord*, *Show*, *Num* y *Fractional*.

2) (2.5 puntos) Usando *foldr*, define una función que devuelva las partes (todas las sublistas) de una lista

```
partes [1, 2, 3] => [ [], [1], [2], [3], [1, 2], [2, 3], [1, 3], [1, 2, 3] ]
```

Nota: no es necesario que los resultados aparezcan en el mismo orden que en el ejemplo.

3) (1.5 puntos) Define usando listas por comprensión la función *concat*.

4) (3 puntos) Define una función *hojas* que devuelva los nodos hoja de un árbol (aquellos que no tienen hijos). Da versiones para árboles binarios y generales.