



NOTA. Responde a cada ejercicio en una hoja diferente.

Ejercicio 1

[3 puntos]

Algunos números naturales cumplen la siguiente propiedad: la suma de cualesquiera dos dígitos adyacentes del número da como resultado un número primo. Por ejemplo, 123 cumple esa propiedad porque $1 + 2 = 3$ y $2 + 3 = 5$, y 3 y 5 son primos.

Se pide: desarrollar un programa en Pascal que muestre por pantalla todos los números que cumplan con esta propiedad y sean menores o iguales que un número natural N introducido por teclado.

Nota: No se permite usar datos de tipo Array ni datos de tipo String.

Ejercicio 2

[3.5 puntos]

Para determinar la puntería de las nuevas armas automáticas que fabrica una empresa se utiliza el siguiente experimento: se prepara una superficie grande como blanco y se procede a efectuar múltiples disparos apuntando a un mismo punto de dicho blanco. Los puntos donde han impactado las balas se miden y se guardan en un fichero secuencial con una estructura fija (ver figura).

Estructura de datos	Ejemplo de ejecución	Figura ilustrativa						
<pre>type tpPunto = record x, y: Real; end; tpImpactos = file of tpPunto;</pre>	<p>Fichero:</p> <table border="1"><tr><td>x = 4</td><td>x = -1</td><td>x = 0</td></tr><tr><td>y = 5</td><td>y = 0</td><td>y = -2</td></tr></table> <p>Salida por pantalla:</p> <p>Centro : 1.00 , 1.00 Radio : 5.00</p>	x = 4	x = -1	x = 0	y = 5	y = 0	y = -2	
x = 4	x = -1	x = 0						
y = 5	y = 0	y = -2						

A partir de esa información se obtiene un círculo que contiene todos los impactos, donde el centro del círculo indica la puntería y el radio del círculo la precisión.

Se pide: desarrollar un programa Pascal que lea el fichero `impactos.dat` de tipo `tpImpactos` y muestre por pantalla el círculo correspondiente: primero el centro del círculo, que será la media de las posiciones de todos los puntos del fichero, y después el radio, que será la distancia desde el centro al impacto más lejano.

Ejercicio 3

[3.5 puntos]

Para un revolucionario algoritmo de navegación GPS, se representan las conexiones por carretera en una región determinada mediante un grafo de distancias $G = (V, E)$. El conjunto de vértices V es el conjunto de lugares clave (ciudades, pueblos, otros lugares de interés) que están identificados mediante un número natural. El conjunto de aristas E son las conexiones mediante carretera, cada una de las cuales está asociada a la distancia correspondiente entre los dos lugares clave conectados (como ilustra la Figura 1). Un grafo de distancias puede representarse mediante una **matriz de distancias** M con una fila y una columna para cada vértice. Dados dos vértices cualesquiera α y β , si están conectados mediante una arista de distancia d , entonces $M[\alpha, \beta] = M[\beta, \alpha] = d$, y si no están conectados (no existe carretera que los conecte) entonces $M[\alpha, \beta] = M[\beta, \alpha] = 0$ (ver Figura 2).

Se pide: Escribir una función en Pascal que reciba como parámetro una matriz de distancias M (que representa un grafo de distancias por carretera) y compruebe si la matriz es correcta o no. Para considerarse correcta, M debe cumplir todas las siguientes condiciones:

- M debe ser cuadrada.
- M debe ser simétrica.
- Todos los elementos de la diagonal principal de M deben ser 0.
- Ningún vértice de M está desconectado; es decir, para todo vértice existe alguna arista que lo conecta con otro vértice.

Se supondrá la siguiente definición de tipos:

type

tpMatrizIncidencia = **record**

numFilas, numColumnas : 1..100;

matriz : **array** [1..100, 1..100] **of** Real;

end;

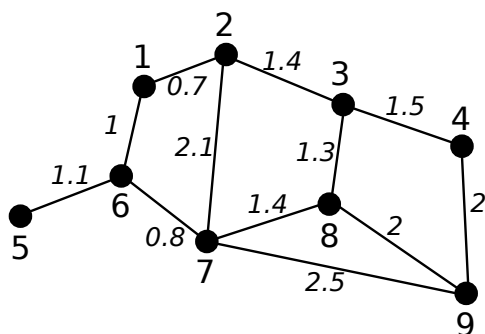


Figura 1: Grafo, mapa de distancias.

	1	2	3	4	5	6	7	8	9
1	0	0.7	0	0	0	1	0	0	0
2	0.7	0	1.4	0	0	0	2.1	0	0
3	0	1.4	0	1.5	0	0	0	1.3	0
4	0	0	1.5	0	0	0	0	0	2
5	0	0	0	0	0	1.1	0	0	0
6	1	0	0	0	1.1	0	0.8	0	0
7	0	2.1	0	0	0	0.8	0	1.4	2.5
8	0	0	1.3	0	0	0	1.4	0	2
9	0	0	0	2	0	0	2.5	2	0

Figura 2: Matriz de distancias para la Figura 1.