

Metodología de la Programación y Algoritmia
Convocatoria de Junio 2014**Apellidos** _____**Nombre** _____ **DNI** _____

1.- El algoritmo de Dijkstra calcula el coste del camino mínimo desde un vértice de un grafo a todos los demás vértices. Una implementación del algoritmo, vista en clase, para calcular el número de caminos desde el vértice 1 a todos los demás es

```
función Dijkstra(C:entero+U{0}[n], P:real+[n,n], D:&real[n])
    i,j,x:natural

    para i←1 hasta n hacer
         $D_i \leftarrow P_{1,i}$ 
    fpara
        para i←1 hasta n-1 hacer
            x ← elegirvértice_costemin(C,D)
             $C_x \leftarrow 0$ 
            para j←1 hasta n hacer
                si pertenece(C,j)
                     $D_j \leftarrow \min(D_j, D_x + P_{x,j})$ 
            fsi
        fpara
    ffunción
```

donde

- C: vector que contiene los números de los vértices, numerados de 1 a n, en las correspondientes posiciones 1,...,n. El vértice desde el cual se parte (en este caso el vértice 1) se le asigna el valor 0.
- P: matriz de adyacencia del grafo.
- D: vector que contendrá las distancias mínimas desde el vértice inicial a cada uno de los vértices del grafo. En la posición 1 contiene la distancia al vértice 1, en la 2 al vértice 2 y así sucesivamente.
- `elegirvértice_costemin(C:entero+U{0}[n], D:real[n]):natural`
Esta función devuelve el número del vértice que tiene menor coste.
C es un vector que contiene los números de los vértices, en el caso de que el valor de un vértice i sea 0 (es decir, $C_i = 0$), el vértice i no se considera para calcular el vértice que tiene menor coste.
D es un vector con los costes asociados a cada uno de los vértices indicados en C.
- `pertenece(C:entero+U{0}[n], i:natural):booleano`
Función que indica si el vértice situado en la posición i del vector C ha sido seleccionado o no. Un vértice ha sido seleccionado si su valor es distinto de 0, es decir $C_i \neq 0$.

Por ejemplo, aplicando el algoritmo el grafo de la página siguiente el resultado es $D = \underline{\quad} 25 \ 21 \ 26 \ 20$, indicando los costes de los caminos más cortos para llegar a los vértices 2, 3, 4 y 5, respectivamente.

Diseña un algoritmo lo más eficiente posible que además de obtener el coste de los caminos más cortos desde el vértice 1, imprima los caminos. Para el grafo del ejemplo el algoritmo imprimirá:

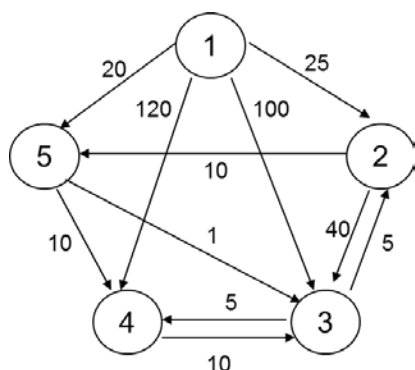
```
1 -> 2
1 -> 5 -> 3
1 -> 5 -> 3 -> 4
1 -> 5
```

Explica cómo funciona tu algoritmo, qué tipo de estrategia sigue y por qué e impleméntalo en pseudocódigo. Puedes utilizar como base el algoritmo de Dijkstra. La función `imprimir(...)` imprime el valor de la variable o dato que se introduce como argumento.

(4.0 puntos)

Metodología de la Programación y Algoritmia

Convocatoria de Junio 2014



Matriz de adyacencia P

	1	2	3	4	5
1	0	25	100	120	20
2	INF	0	40	INF	10
3	INF	5	0	5	INF
4	INF	INF	10	0	INF
5	INF	INF	1	10	0

INF: valor muy grande indicando que no hay arista.

2.- El Ayuntamiento de Elche va a organizar unas jornadas de fútbol para los Institutos de Educación Secundaria. Cada centro puede participar con un solo equipo formado por alumnos del centro. En estas jornadas cada equipo jugará 1 partido contra cada equipo participante.

Las jornadas empezarán el 1 de Julio y se celebrarán partidos diariamente. Cada equipo debe jugar un partido por día y puede descansar como mucho 1 día durante toda la duración del evento. Los partidos se jugarán por la tarde a partir de las 20:00 en las instalaciones deportivas de los centros.

2.a) Explica **detalladamente** el funcionamiento de un algoritmo que obtenga una distribución de partidos que cumpla las restricciones del problema para cualquier número de centros participantes e indica de qué tipo de estrategia se trata y por qué.

(2.0 puntos)

2.b) Se han inscrito los siguientes 9 Institutos de Educación Secundaria: Carrús, Cayetano Sempere, L'Assumpció, Misteri d'Élx, Nit de l'Albá, Pere Ibarra, Sixto Blanco, Tirant Lo Blanc y Victoria Kent.

Aplica el algoritmo a este caso, indicando los valores que se van asignando de distribuciones de partidos hasta llegar a la distribución final de partidos.

(1.5 puntos)

3.- Dado el siguiente algoritmo

```

función junio14(a:entero, b:entero):entero
    si a < 3
        devolver a + b
    si no
        devolver a + junio14(a / 3, b - 2)
    fsi
función

```

3.a) Indica de qué tipo de recursividad se trata y por qué.

(0.5 puntos)

3.b) Calcula la complejidad asintótica y justifica tu respuesta.

(1.0 puntos)

3.c) Convierte el algoritmo a iterativo.

(1.0 puntos)