

Nombre y Apellidos: \_\_\_\_\_

dni: \_\_\_\_\_

Titulación y Grupo: \_\_\_\_\_

## PROGRAMACIÓN DINÁMICA

1. (5 ptos) Los Reptilianos nos invaden. Están intentando localizar desde donde retransmitimos para así destruir a la resistencia. Para evitarlo, nuestro líder supremo, Subiris, “el magnífico”, nos ha ordenado que las transmisiones sean con el menor número de bits posible. Como oficial de comunicaciones de la resistencia del sector Al-Andalus, su labor es la de gestionar las comunicaciones de la manera más eficiente posible, evitando que lo localicen y ser aniquilado por un rayo Gamma.

Supongase que se quiere transmitir un mensaje  $\mathbf{S} = s_0 \cdot \dots \cdot s_{m-1}$  dado como una cadena de  $\mathbf{m}$  símbolos. A tal fin, dispone de  $\mathbf{r}$  codificaciones distintas (ASCII, binario, punto flotante, etc...). Sea  $\mathbf{B}_{i,j}$  el número de bits necesarios para codificar el  $i$ -ésimo símbolo en el  $j$ -ésimo código ( $\mathbf{B}$  es por lo tanto de tamaño  $\mathbf{m} \times \mathbf{r}$ ). Inicialmente el transmisor del condensador de flujo encargado de transmitir el mensaje está fijado al código 0 pero puede ser cambiado durante la transmisión cuando se desee. Para ello se necesita enviar un código de control compuesto de  $\mathbf{C}_{ij}$  bits si se desea cambiar del código  $i$  actual al código  $j$  ( $\mathbf{C}_{r \times r}$  y  $\forall i \in [0, \mathbf{r} - 1] : \mathbf{C}_{ii} = \mathbf{0}$ ). Su objetivo es determinar cómo enviar el mensaje empleando el número mínimo de bits. ¡Larga vida a Subiris!

Su suboficial de comunicaciones, Neo, le ha comunicado en una reunión *brainstorming* que el objetivo a encontrar sería una función  $MNB(i, c)$  que indicaría el Mínimo Número de Bits necesario para retransmitir el mensaje desde el carácter  $i$  en adelante, teniendo en cuenta que actualmente se encuentra en el código  $c$ , decidiendo en cada paso en qué código retransmitir el siguiente carácter, estando así la solución del problema en  $MNB(0, 0)$ . Quizás tenga razón...

- a) (2 ptos) Expresar mediante una ecuación de Bellman el número óptimo de bits ( $MNB(i, c)$ ).

$$MNB(i, c) = \begin{cases} i = m - 1 & \min_{0 \leq k < r} (C_{c,k} + B_{i,k}) \\ \text{en otro caso} & \min_{0 \leq k < r} (C_{c,k} + B_{i,k} + MNB(i + 1, k)) \end{cases}$$

- b) (0,5 pto) Hacer un análisis de como seria la estructura de datos que almacena los resultados parciales, así como donde se encuentra la solución, como ha de recorrer la estructura, tamaño de esta y su complejidad espacial
- (0,1 pto) La estructura de datos será una matriz de  $m$  filas (he considerado que  $i=m-1$  es el ultimo caracter a codificar) y  $r$  columnas (desde el código 0 al código  $r-1$ ) .
  - (0,1 pto) La solución se encuentra en la posición  $MSB(0,0)$  ya que empiezo con el caracter 0 en el código 0 y ahí se encuentra la solución de encontrar el Mínimo Número de Bits necesario para retransmitir el mensaje desde el carácter 0 en adelante, teniendo en cuenta que actualmente se encuentra en el código 0.
  - (0,1 pto) Para encontrar la posición  $MNB(i,c)$  necesito todos los elementos de la fila justamente de debajo ( $i+1$ ), por lo que debería recorrer la tabla de abajo a arriba y luego, indiferentemente de izquierda a derecha o de derecha a izquierda. La fila 0 no seria necesaria recorrerla entera ya que solo necesitamos la posición  $(0,0)$
  - (0,1 pto) Los casos base serian los que se encuentran justamente en la ultima fila de la tabla, donde se esta decidiendo con que código retransmito el ultimo mensaje
  - (0,1 pto) La complejidad espacial es por tanto  $32 \cdot m \cdot c$  bits
- c) (1,5 pto) Implementar un algoritmo de programación dinámica en función de la anterior ecuación que devuelva el número de bits mínimo ( $MNB(0,0)$ )
- La solución puede verse en la figura 1
- d) (1,5 pto) Añada el código necesario que devuelva en que codificación ha de retransmitirse cada carácter ( $int[] sol$ , donde  $\forall k \in [0, m-1] : sol[k] \in [0, r-1]$ )
- La solución puede verse en la figura 1

## ALGORITMOS ÁVIDOS

1. (2 ptos) Una misteriosa arma biológica ha sido desarrollada por los Reptilianos en la batalla por la tierra. El sector Al-Andalus esta en peligro. Tras muchos estudios, nuestra científica jefe, Niobe, se ha dado cuenta que sigue un patrón de contagio múltiple: solo si un individuo que está conectado con al menos  $k$  individuos infectados, pasa a estar infectado él también. No quieras saber los síntomas. Subiris junto con sus generales (Morfeo y Trinity) ha decidido eliminar a las personas que potencialmente pueden ser infectadas, mandándolas a luchar a primera linea de batalla, pero solo las necesarias. Cada población de este sector ha sido modelado con un grafo con  $n$  vértices donde cada uno de los cuales representa un

```

public static int[] codificación(String S,int[][] B, int[][] C) {
    /*
     * Implementar un algoritmo de programación dinámica en
     * función de la anterior ecuación que devuelva el número de bits mínimo
     * (MNB(0, 0))
     */
    int r = C.length;
    int m = S.length();
    int[][] MNB= new int[m][r];

    for (int i = m-1; i<=0 ; i--) {
        for (int c = 0; c<r ; c++) {
            if (i==m-1) {
                MNB[m-1][c]=C[c][0]+B[m-1][0]; //k==0
                for (int k=1; k < r; k++) {
                    MNB[m-1][c]= Math.min(MNB[m-1][c],
                                            C[c][k]+B[m-1][k]);
                }
            }
            else {
                MNB[i][c]=C[c][0]+B[i][0] + MNB[i+1][0]; //k==0
                for (int k=1; k < r; k++) {
                    MNB[i][c]= Math.min(MNB[i][c],
                                        C[c][k]+B[i][k]+ MNB[i+1][k]);
                }
            }
        }
    }

    /*
     * Añada el código necesario que devuelva en que codificación
     * ha de retransmitirse cada carácter
     */
    int[] sol = new int[r];
    int i = 0;
    int c = 0;
    while (i<m-1) {
        int k = 0;
        while (MNB[i][c]!=C[c][k]+B[i][k]+ MNB[i+1][k]) k++;
        sol[i]=k;
        c=k;
        i++;
    }
    int k = 0;
    while (MNB[m-1][c]!=C[c][k]+B[m-1][k]) k++;
    sol[m-1]=k;

    return sol;
}

```

Figura 1: Codificación ejercicios 1.c y 1.d

habitante de la población, y en el que las aristas representan contacto físico entre individuos. El objetivo es, dado un grafo  $G(V, E)$  que representa las interacciones entre los individuos de una población y la función  $\text{grado}(G, u) = k$ , donde  $k$  es el número de vértices de  $V$  que están relacionados con  $u$  con alguna arista, determinar qué individuos deben ser enviados al frente de manera que podamos seguir luchando para salvar a nuestro planeta. ¡Larga vida a Subiris!

a) (2 pts) Definir un algoritmo voraz para resolver el problema.

- $S = (L, G(V, E))$  donde  $L = \{l_i\}$ ,  $l_i \in V = j \equiv$  el  $i$ -ésimo candidato a ir a filas es el individuo  $j$
- $S_{ini} = (\{ \}, G(V, E))$
- $\text{Final}(S) = \nexists u \in V : \text{grado}(u) \geq k$
- $\text{Valida}(S, t) = t \in V \wedge t \notin L$
- $\text{Seleccion}((L, G(V, E))) = (L + \{t\}, G'(V', E')) : \text{Valida}(S, t) \wedge \text{grado}(G, t) = \alpha \wedge \nexists u \in V : \text{grado}(G, u) > \alpha \wedge V' = V - \{t\} \wedge E' = E - \{(u, v) \in E : u = t \vee v = t\}$

2. (2,5 pts) Nuestra logística esta debilitada y debemos encontrar el camino más corto para enviar suministros a otros sectores desde Al-Andalus. Sea el grafo que se muestra a continuación, en el cual se representa cada uno de los sectores que estamos defendiendo actualmente como vértices del mismo, y las aristas (ponderadas) el coste de enviar munición a ese sector desde otro sector adyacente. Nuestro gestor jefe, Cifra, quiere usar el algoritmo de Edsger Dijkstra (11 de mayo de 1930 (Rotterdam) - 6 de agosto de 2002 (Nuenem)) para obtener el camino mínimo desde nuestro sector (sector a) al resto de sectores. Subiris le ha pedido a Cifra (que se lo pide a usted) que le indique en la siguiente tabla cual seria el resultado. ¡Larga vida a Sub...! Subiris fue enviado al frente tras el ejercicio anterior

Aristas del árbol	Lista ordenada de Aristas
$a(-, 0)$	$b(a, 3), d(a, 7), c(-, \infty), e(-, \infty)$
$a(-, 0)$ $b(a, 3)$	$d(a, 5), c(b, 7), e(-, \infty)$
$a(-, 0)$ $b(a, 3)$ $d(a, 5)$	$c(b, 7), e(d, 9)$
$a(-, 0)$ $b(a, 3)$ $d(a, 5)$ $c(b, 7)$	$e(d, 9)$
$a(-, 0)$ $b(a, 3)$ $d(a, 5)$ $c(b, 7)$ $e(d, 9)$	

