**ALGORÍTMICA**

**2º Grado en Ingeniería Informática**

**Examen de la convocatoria ordinaria. Curso 2017-2018**

1. (1 punto) Describa cómo calcular el tamaño del caso, halle la ecuación en recurrencias y proporcione la eficiencia en el caso peor para el código siguiente:

void F(int a, int b) {

for (int i= a; i<b; i++)

F(a, a+(b-a)/2);

}

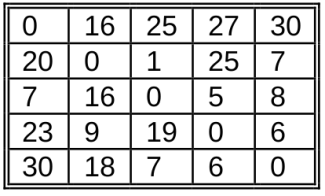
1. (1 punto) Describa el algoritmo Quicksort y en qué circunstancia se da su caso peor. Calcule la eficiencia del algoritmo en el peor caso y en el mejor caso. A la vista de los resultados que obtenga, justifique porqué se le considera como el algoritmo de ordenación más rápido.
2. Un estudiante tiene D días para estudiar un examen que consta de N temas. Se estudia los temas completos, y en estudiar un tema *i* calcula que tardará un tiempo de *ti* días enteros. Suponiendo que la probabilidad de que caiga cada tema *i* en el examen (frente a que no caiga) es *pi*, el alumno desea seleccionar qué temas estudiarse en el tiempo que tiene de modo que maximice la suma de las probabilidades de que los temas estudiados caigan en el examen. Responda a las siguientes cuestiones:
   1. (0,5 puntos) Plantee formalmente el problema.
   2. (1 punto) ¿Es posible resolver este problema de forma óptima con Programación Dinámica? En caso afirmativo, indique cómo cumple las condiciones necesarias para aplicar la técnica y explique detalladamente la ecuación en recurrencias resultante. En caso negativo, indique qué condición/es no cumple el problema para aplicar la técnica.
   3. (1 punto) Diseñar otra alternativa para solucionar el problema con un enfoque Greedy cuya idea general sea “Estudiar primero los temas más probables”. Demuestre su optimalidad o no optimalidad para el siguiente caso:

Número de días para el examen D= 4; Número de temas N=5.

Probabilidades P=(p1, p2, p3, p4, p5)= (100%, 50%, 50%, 80%, 60%)

Tiempo de estudio T= (t1, t2, t3, t4, t5)=(3, 2, 1, 2, 1)

1. Un robot tiene que soldar 5 puntos defectuosos X={1, 2, 3, 4, 5} en un circuito. El coste de moverse de un punto defectuoso *i* a otro *j* viene dado por la matriz indicada al final del ejercicio (i=fila, j=columna). Se desea construir un algoritmo que indique el orden en el que se tienen que soldar los puntos de modo que el coste total del movimiento sea mínimo. Suponiendo que comienza a soldar desde la primera soldadura (i=1),
   1. (hasta 1,5 puntos) Desarrolle un algoritmo utilizando alguna técnica de exploración en grafos que permita obtener la secuencia de puntos a soldar con mínimo coste total.
   2. (1 punto) Explique el funcionamiento del algoritmo sobre el ejemplo de la matriz siguiente:



**Tiempo de examen total: 2 horas 30 minutos.**