

**Prácticas de Algorítmica.**  
**3º de Grado en Ingeniería Informática.**  
**Curso 2021-2022.**  
**Práctica 6.**

**Objetivos.**

Con esta práctica se pretende que el alumno implemente ejemplos de algoritmos basados en el Backtracking y de algoritmos probabilistas.

Implementad el algoritmo del problema de las  $n$  reinas, siendo  $n$  un valor introducido por el usuario. El programa tendrá tres opciones.

1. Parte obligatoria (**Nota máxima: 7.5**)

1. Mediante el método del Backtracking. Las soluciones serán guardadas en una estructura de datos adicional que será implementada por el alumno, y después, usando dicha estructura, se mostrarán todas las soluciones (**3 puntos**).
2. Tomando como base la opción anterior, implementar una opción en la que se obtenga una única solución. Esta solución ha de colocar obligatoriamente la reina de la primera fila en la columna central ( $c/2$ ) y se ha de medir el tiempo que tarda en obtenerse (**2 puntos**).
3. Mediante el algoritmo de las Vegas visto en clase. Para este caso hay que mostrar obligatoriamente una solución indicando el número de intentos que se han realizado para obtener esa solución y el tiempo que se ha consumido (**2.5 puntos**).

2. Parte opcional (**Nota máxima: 10**).

1. Suponiendo que el colocar una reina implica un coste igual al producto de la fila por la columna que ocupa (la reina fila 3, columna 4, implica un coste de 12), buscar de entre todas las soluciones del problema, cual es la que genera un menor coste aplicando un algoritmo de poda al algoritmo de búsqueda de todas las soluciones implementado anteriormente (**1.25 puntos**).
2. Implementar el algoritmo de Montecarlo para verificar el producto de matrices. Para ello proceder de la siguiente forma:
  1. Generar la matrices cuadradas A y B de forma aleatoria rellenándolas con valores que serán  $\{-1, 0, 1\}$  con un orden  $n$  introducido por el usuario.
  2. Multiplicar A por B y como resultado obtendremos la matriz AB. Medir el tiempo que se tarda en multiplicarlas.
  3. Obtener la matriz C de la siguiente forma: se lanza un dado y si sale un número par, la matriz C será igual que la AB; si sale impar, se obtendrá la matriz C a partir de la matriz AB sumándole 1 al elemento  $AB(n/2, n/2)$  (el resto de los elementos no se toca)
  4. Generar el vector X de  $n$  elementos relleno de valores que serán 0 ó 1, con la misma probabilidad.
  5. Comprobar si se verifica el producto realizando el número de pruebas que indique el usuario. En el caso de que el producto no se verifique, se indicará el número de pruebas que se han necesitado. En caso contrario se indicará cual es la probabilidad de que el producto se verifique. La probabilidad será  $1 - (1/2)^{nPruebas}$ . También hay que medir el tiempo que se tarda en esas pruebas para compararlo con el tiempo que se tardó en realizar el producto.

(**1.25 puntos**)

**Fecha de comienzo: 23 de noviembre de 2019.**

**Fecha máxima de entrega: 10 de Diciembre de 2019.**