

Ejercicios Temas 4 y 5

Algoritmia para problemas difíciles

Departamento de Informática e Ingeniería de Sistemas
 Escuela de Ingeniería y Arquitectura – Universidad de Zaragoza

9 de diciembre de 2015

Ejercicios sobre heurísticas

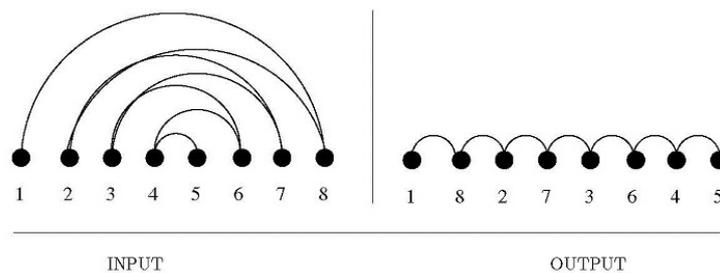
Nota importante: En todos los ejercicios de esta hoja se valorará la implementación y prueba experimental de las soluciones.

Ejercicio 1 Un *desarreglo* es una permutación p de $\{1, \dots, n\}$ tal que ningún elemento está en su sitio, es decir, $p_i \neq i$ para todo $1 \leq i \leq n$. Diseñar e implementar una heurística de tipo *simulated annealing* que obtenga soluciones razonables para el problema de encontrar uno o todos los desarreglos de n elementos.

Ejercicio 2 En el problema de *reconstrucción de puntos a partir de distancias*, nos dan un multiconjunto A de $n(n-1)/2$ distancias positivas y se trata de encontrar una colocación de n puntos para obtener esas distancias. Es decir, si los n puntos son x_1, \dots, x_n con $x_i \leq x_{i+1}$ entonces $A = \{x_i - x_j \mid j < i\}$. Por ejemplo las distancias $\{2, 2, 4, 4, 6, 8\}$ puede corresponder a los puntos $x_1 = 1, x_2 = 3, x_3 = 5, x_4 = 9$. Diseñar e implementar una heurística de tipo *simulated annealing* que obtenga soluciones razonables para el problema de reconstrucción de puntos a partir de distancias.

Ejercicio 3 Un *multiconjunto* puede tener elementos repetidos. Un multiconjunto de n elementos puede tener menos de $n!$ permutaciones (por ejemplo $\{1, 1, 2, 2\}$ sólo tiene 6 permutaciones). Diseñar una heurística que obtenga soluciones razonables para el problema de construir todas las permutaciones de un multiconjunto.

Ejercicio 4 Diseñar e implementar una heurística de tipo *simulated annealing* que obtenga soluciones razonables para el problema de minimización de anchura de banda o *Bandwidth* definido como sigue. Dado un grafo, ¿qué permutación p de los vértices minimiza la “longitud dibujada” de las aristas cuando los vértices se ordenan sobre una línea recta, es decir, minimiza $\max_{\text{arista } (i,j)} |p(i) - p(j)|$?



Ejercicio 5 Diseñar e implementar una heurística de tipo *simulated annealing* que obtenga soluciones razonables para el problema de MAXSAT definido como sigue. Nos dan un conjunto de cláusulas, y queremos encontrar una asignación que satisface el mayor número posible de ellas.

Ejercicio 6 Diseñar e implementar una heurística de tipo *simulated annealing* que obtenga soluciones razonables para el problema de MAXCLIQUE definido como sigue. Dado un grafo G , ¿cuál es el mayor conjunto de vértices S tal que para todo $u, v \in S$ existe la arista (u, v) en G ?

Ejercicio 7 Encontrar todos los números entre 0 y 255 que tienen una representación binaria tal que el número total de ceros es igual al de unos. Por ejemplo, 240 es uno de esos números (su representación es 11110000) y 153 es otro (representación: 10011001).

Discutir un algoritmo para encontrar esos números de manera directa y exhaustiva.

Discutir un algoritmo genético para encontrar números que cumplan la condición. ¿Los encuentra todos? Si es así, ¿cómo? Si no, ¿qué habría que hacer para encontrar más números que cumplan la condición?

Ejercicio 8 Discutir la posibilidad de resolver sudokus mediante algoritmos genéticos.

Ejercicio 9 Diseñar e implementar una heurística de tipo *simulated annealing* que obtenga soluciones razonables para el problema de Set Cover (transparencia 40 de Algoritmos Aproximados).

Ejercicio 10 Diseñar e implementar una heurística de tipo *simulated annealing* que obtenga soluciones razonables para el problema de Máximo Subgrafo Acíclico definido como sigue. Dado un grafo G , ¿cuál es el conjunto más pequeño de vértices cuyo borrado deja el grafo acíclico?

En caso de entregar alguno de estos ejercicios, la fecha límite es el jueves 7 de enero.
--

Antes de realizar cualquiera de estos ejercicios el alumno debe enviar un correo a elvira@unizar.es indicando qué ejercicio desea realizar.

Cualquier fuente utilizada en la resolución de estos ejercicios debe ser indicada claramente en la solución.
--