

Optimización - 2013 - Práctica 4
Backtracking, Branch & Bound

1. Se tienen K, a_1, \dots, a_n enteros no negativos.
 - a) Implementar en Python un algoritmo usando la técnica de Backtracking para determinar si existe un subconjunto de los $\{a_i\}$ tales que sumen exactamente K .
 - b) Escribir un algoritmo Branch & Bound para determinar el subconjunto de mayor suma tal que la misma sea $\leq K$.
2. Escribir un algoritmo mediante la técnica de Backtracking para poder determinar todos los cliques (subgrafo completo maximal) de un grafo.
3. Escribir un algoritmo Branch & Bound para el problema Programación Lineal Entera.
4. Se tiene un tablero de ajedrez de $n \times n$ casilleros, se busca colocar n damas de manera tal que no se amenacen entre sí. Diseñar un algoritmo con la técnica de Backtracking para resolver este problema e implementarlo en Python.
5. Implementar en Python el algoritmo de Backtracking Recursivo visto en clase que resuelve el Sudoku.