**Laboratorio Nro. 02  
Fuerza bruta**

|  |  |  |
| --- | --- | --- |
| **Tomás Bernal Zuluaga**  Universidad Eafit  Medellín, Colombia  tbernalz@eafit.edu.co | **Andrés Prada**  Universidad Eafit  Medellín, Colombia  aprada@eafit.edu.co | **Damián Duque**  Universidad Eafit  Medellín, Colombia  daduqueg@eafit.edu.co |

**3) Simulacro de preguntas de sustentación de Proyectos**

**3.1** El código lo que hace es crear un método que recibe un grafo y una lista y usando el método permutations de la clase itertools para conseguir todas las permutaciones posibles de la lista, esto devolviendo todas las posibles permutaciones y sacando el peso de cada permutación, este peso se compara con un entero inicializado en inf y asi se sigue comparando hasta pasar por todas las posibles permutaciones y al final dejando el menor peso posible.

**3.2** V!\*2E

**3.3** No es posible por los medios que tenemos además el (n!) es un numero exageradamente grande 3.0414093e+64  
Y esto en años serian 8.3555201e+61

**3.4** El código se encarga de ir haciendo backtracking con base en las posiciones en las que la reina no podía estar. La estructura de datos empleada fueron los Array (una y dos dimensiones, la primera para guardar las posibles respuestas, y la segunda para identificar donde no podía estar las reinas.

**3.5** O(n2)

**3.6** n es el número de filas que se van a digitar

***4) Simulacro de Parcial***

**4.1.1** actual > máximo

**4.1.2** O(n2), donde n es el arreglo a

* 1. n/
  2. A

**4.3.1** i -m

**4.3.2** n

**4.3.3** O(n\*m)

**4.4.1** temp%10

**4.4.2** D

**4.5.1** i+1

**4.5.2** right== left