**Descrição do Problema e da Solução**

O objetivo do problema é desenvolver um programa para distribuir o máximo possível de brinquedos de Natal por T crianças. O programa recebe N fábricas distribuídas por M países, com cada fábrica produzindo um único brinquedo com um determinado stock máximo. Cada criança recebe apenas um brinquedo desejado. Cada país tem um mínimo de prendas distribuídas e um máximo de prendas que podem ser exportadas.

A solução passa por criar dois conjuntos de variáveis:

1. O primeiro conjunto representa o número de crianças de cada país que irá receber uma prenda, ligando, assim, os países às crianças lá residentes.
2. O segundo conjunto denota o número de brinquedos que cada criança recebe, ligando, assim, as crianças às fábricas que produzem os brinquedos que cada uma deseja.

Ambos os conjuntos são binários, pois têm peso 0 ou 1. Além disso, temos de colocar a função objetivo que é maximizar o número de crianças que irá receber uma prenda. Temos ainda de modelar as restrições do nosso problema:

1. O número de prendas que uma criança de um país recebe é igual ao número de prendas fornecidas pelas fábricas desejadas por esta criança.
2. O número de prendas fornecidas por uma fábrica tem de ser menor que o seu stock.
3. A soma das prendas dadas por fábricas de um país a crianças de outros países não pode ultrapassar o limite de exportação desse país.
4. Cada país tem de receber um determinado mínimo de prendas.

**Análise Teórica**

* **Leitura dos dados de entrada e processamento dos dados:** simples leitura do input, com ciclos a depender de N, de M e de T. Em cada ciclo adiciona-se dados a dicionários com a complexidade O(), com a exceção do último ciclo em que se adiciona uma lista a um dicionário, cuja complexidade no pior caso é O(). Deste modo, a complexidade final será O(). Logo, O().
* **Criação das variáveis e restrições:** para criar o conjunto de variáveis País-Criança, percorre-se um ciclo a depender de T. Assim, O(). Para criar o conjunto de variáveis Criança-Fábrica, percorre um ciclo a depender de T com outro a depender de N, assim O(). Para criar a função objetivo, percorre-se outro ciclo a depender de T, logo O(). As quatro restrições têm complexidades O(), O(), O() e O(). Logo, O().
* **Complexidade global da solução:** deste modo, a solução tem uma complexidade global de O(), redutível a O().

**Avaliação Experimental dos Resultados**

Foram realizados 27 testes com valores de N entre 100 e 300, com incrementos de 100, valores de M entre 10 e 30, com incrementos de 10, e para os valores de T, entre 1000 e 3000, com incrementos de 1000. Para além disso, foram usados valores constantes para a variância, capacidade máxima e pedidos máximos, sendo eles 0.5, 50 e 5.

Ao analisar o gráfico, notou-se que a relação não era linear, pelo que se usou uma complexidade efetiva de O(), de forma a obter um gráfico mais linear.

Uma imagem com captura de ecrã, Gráfico, file, texto

Descrição gerada automaticamente

\*Testes corridos num Macbook M1 Pro com 16GB de RAM

Uma imagem com texto, captura de ecrã, número, Tipo de letra

Descrição gerada automaticamente

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N** | **M** | **T** | **NMT^3** | **T (s)** |
| 100 | 10 | 1000 | 1E+12 | 0,15 |
| 100 | 20 | 1000 | 2E+12 | 0,15 |
| 200 | 10 | 1000 | 2E+12 | 0,15 |
| 300 | 10 | 1000 | 3E+12 | 0,16 |
| 100 | 30 | 1000 | 3E+12 | 0,16 |
| 200 | 20 | 1000 | 4E+12 | 0,15 |
| 300 | 20 | 1000 | 6E+12 | 0,16 |
| 200 | 30 | 1000 | 6E+12 | 0,17 |
| 100 | 10 | 2000 | 8E+12 | 0,31 |
| 300 | 30 | 1000 | 9E+12 | 0,16 |
| 100 | 20 | 2000 | 1,6E+13 | 0,28 |
| 200 | 10 | 2000 | 1,6E+13 | 0,3 |
| 300 | 10 | 2000 | 2,4E+13 | 0,26 |
| 100 | 30 | 2000 | 2,4E+13 | 0,28 |