

Benemérita Universidad Autónoma De Puebla

Análisis y Diseño De Algoritmos

Integrantes

López Sánchez Axel Yeray

López Bonilla Nairobi Yadira

Pérez García Iván

Mauricio Gutiérrez Dulce María

**Tarea de investigación en equipo: reporte de
investigación del problema de las jarras**



Título: El problema de las jarras

Autores:

Dijkstra, Edsger W. - Contribuyó al desarrollo de algoritmos, incluido el famoso algoritmo de Dijkstra para encontrar el camino más corto en un grafo.

Knuth, Donald E. - Autor de la serie de libros "The Art of Computer Programming" y conocido por sus contribuciones a la teoría de algoritmos.

Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L., y Stein, Clifford - Autores del libro "Introduction to Algorithms" (Introducción a los algoritmos), que es una referencia estándar en el campo de la ciencia de la computación y algoritmos.

Desarrollo:

```
def resolver_jarras(jarra1, jarra2, objetivo, historial_acciones):
```

```
    # Verificar si hemos alcanzado el objetivo
```

```
    if jarra1 == objetivo or jarra2 == objetivo:
```

```
        return historial_acciones
```

```
    # Aplicar todas las acciones posibles (llenar, vaciar, verter)
```

```
    acciones_posibles = ["Llenar Jarra 1", "Llenar Jarra 2", "Vaciar Jarra 1", "Vaciar Jarra 2", "Verter de 1 a 2", "Verter de 2 a 1"]
```

```
    for accion in acciones_posibles:
```

```
        nueva_jarra1, nueva_jarra2 = aplicar_accion(jarra1, jarra2, accion)
```

```
        # Verificar si la nueva configuración ya se ha explorado
```

```
        if (nueva_jarra1, nueva_jarra2) not in historial_acciones:
```

```
            # Llamada recursiva para explorar la nueva configuración
```

```
resultado = resolver_jarras(nueva_jarra1, nueva_jarra2, objetivo,  
historial_acciones + [(nueva_jarra1, nueva_jarra2, accion)])
```

```
# Si se encuentra una solución, retornar el historial de acciones
```

```
if resultado:
```

```
    return resultado
```

```
# Si no se encuentra ninguna solución
```

```
return None
```

```
def aplicar_accion(jarra1, jarra2, accion):
```

```
    # Implementar la lógica para aplicar cada acción
```

```
    # Por ejemplo, llenar jarra, vaciar jarra, verter de una jarra a otra, etc.
```

```
    # Devolver las nuevas capacidades de las jarras después de la acción
```

```
# Uso del algoritmo
```

```
historial_solucion = resolver_jarras(0, 0, 4, [])
```

```
print(historial_solucion)
```

Ejemplos: Ejemplo 1: Jarras de 3 y 5 litros, objetivo: 4 litros

1. Llenar la jarra de 5 litros. (5L, 0L)
2. Verter agua de la jarra de 5 litros a la de 3 litros. (2L, 3L)
3. Vaciar la jarra de 3 litros. (2L, 0L)
4. Verter agua de la jarra de 5 litros a la de 3 litros. (0L, 2L)
5. Llenar la jarra de 5 litros. (5L, 2L)
6. Verter agua de la jarra de 5 litros a la de 3 litros hasta llenarla. (4L, 3L)

Ejemplo 2: Jarras de 4 y 7 litros, objetivo: 5 litros

1. Llenar la jarra de 7 litros. (7L, 0L)

2. Verter agua de la jarra de 7 litros a la de 4 litros. (3L, 4L)
3. Vaciar la jarra de 4 litros. (3L, 0L)
4. Verter agua de la jarra de 7 litros a la de 4 litros. (0L, 3L)
5. Llenar la jarra de 7 litros. (7L, 3L)
6. Verter agua de la jarra de 7 litros a la de 4 litros hasta llenarla. (6L, 4L)
7. Verter agua de la jarra de 4 litros a la de 7 litros. (2L, 6L)
8. Vaciar la jarra de 4 litros. (2L, 0L)
9. Verter agua de la jarra de 7 litros a la de 4 litros. (0L, 2L)
10. Llenar la jarra de 7 litros. (7L, 2L)
11. Verter agua de la jarra de 7 litros a la de 4 litros hasta llenarla. (5L, 4L)

Discusión:

1. Introducción

El problema de las jarras es un clásico desafío en el ámbito de la teoría de grafos y la resolución de problemas. Este consiste en encontrar la manera más eficiente de medir una cantidad específica de líquido utilizando jarras de capacidades diferentes. En este reporte, se analizarán las características fundamentales del problema, así como posibles estrategias de solución.

2. Descripción del Problema

El problema involucra jarras de distintas capacidades, donde el objetivo es medir una cantidad específica de líquido utilizando estas jarras. Cada jarra puede ser llenada, vaciada o volcada en otra jarra. El reto radica en determinar la secuencia óptima de acciones para alcanzar la cantidad deseada de líquido.

3. Parámetros del Problema

Capacidades de las Jarras: Se tienen jarras con capacidades específicas, por ejemplo, una jarra de 3 litros y otra de 5 litros.

Objetivo: La cantidad específica de líquido que se desea obtener.

4. Estrategias de Resolución

Existen diversas estrategias para abordar el problema de las jarras, entre las cuales se encuentran:

Búsqueda Exhaustiva: Explorar todas las posibles combinaciones de acciones hasta encontrar la solución. Este enfoque garantiza encontrar la respuesta, pero puede ser ineficiente en términos computacionales.

Algoritmos de Búsqueda Heurística: Utilizar técnicas como búsqueda en amplitud o búsqueda en profundidad con heurísticas para reducir el espacio de búsqueda y mejorar la eficiencia del algoritmo.

Programación Dinámica: Descomponer el problema en subproblemas más pequeños y resolverlos de manera óptima, almacenando las soluciones intermedias para evitar recálculos.

5. Implementación Práctica

La implementación práctica del problema de las jarras implica la codificación de las acciones posibles (llenar, vaciar, verter) y la aplicación de las estrategias de resolución mencionadas anteriormente. Es crucial considerar la eficiencia y complejidad computacional al seleccionar un método de resolución.

6. Conclusiones

El problema de las jarras, aunque aparentemente simple, presenta desafíos interesantes desde el punto de vista algorítmico. La elección de la estrategia de resolución dependerá de la complejidad del problema y de los recursos computacionales disponibles. La comprensión profunda de este problema proporciona una base valiosa para abordar otros desafíos en el ámbito de la teoría de grafos y la optimización.