El código resuelve un problema de asignación de tareas con duraciones y dependencias utilizando la biblioteca OR-Tools de Google.

1. \*\*Carga de la Biblioteca OR-Tools:\*\*

- Se carga la biblioteca OR-Tools utilizando `Loader.loadNativeLibraries();`.

2. \*\*Obtención de Datos de Entrada:\*\*

- Se obtienen los siguientes datos de entrada desde campos de texto o similar:

- `numTareas`: El número de tareas.

- `numRecursos`: El número de recursos.

- `duracionesStr`: Las duraciones de las tareas como una cadena de texto separada por comas.

- `dependenciasStr`: Las dependencias entre tareas como pares de números enteros separados por comas.

3. \*\*Creación del Array de Dependencias:\*\*

- Se verifica que la longitud de `dependenciasStr` sea un múltiplo de 2 para asegurarse de que los pares estén completos.

- Se crea un array bidimensional llamado `dependencias` para almacenar las dependencias. Los pares de dependencias se extraen de `dependenciasStr` y se almacenan en este array.

4. \*\*Creación del Solucionador de Programación Lineal Entera:\*\*

- Se crea un solucionador de programación lineal entera (`MPSolver`) llamado "AsignacionTareas" con el tipo de problema "CBC\_MIXED\_INTEGER\_PROGRAMMING".

5. \*\*Definición de Variables de Decisión:\*\*

- Se crean variables de decisión llamadas `asignacionTareas` que representan la asignación de cada tarea a un recurso. Estas variables son binarias (0 o 1) y se crean en una matriz.

6. \*\*Definición de la Función Objetivo:\*\*

- Se define la función objetivo para minimizar la duración total del proyecto. Los coeficientes de la función objetivo se configuran en función de las duraciones de las tareas.

7. \*\*Agregación de Restricciones de Dependencia:\*\*

- Se agregan restricciones para garantizar que las dependencias entre tareas se cumplan. Se recorre el array `dependencias` y se crea una restricción para cada dependencia. Las restricciones están relacionadas con las variables de asignación de tareas.

8. \*\*Resolución del Problema:\*\*

- Se resuelve el problema de programación lineal entera utilizando el solucionador.

- Se verifica si se encontró una solución óptima.

9. \*\*Impresión de Resultados:\*\*

- Si se encontró una solución óptima, se muestra la planificación óptima de asignación de tareas y la duración total del proyecto. Esto se hace recorriendo las variables de asignación de tareas y sumando las duraciones de las tareas asignadas a cada recurso.

- Si no se encontró una solución óptima, se muestra un mensaje indicando que no se encontró una solución óptima.

El código obtiene datos de entrada, crea un modelo de programación lineal entera, define variables de decisión y restricciones, resuelve el problema y muestra la solución óptima, si es posible. El resultado es la asignación óptima de tareas a recursos y la duración total del proyecto.