

Instalando Cámaras de Seguridad

NOTA: Si usted está leyendo este documento sin haber extraído el compactado que se le entregó, ciérrelo ahora, extraiga todos los archivos en el escritorio, y siga trabajando desde ahí. Es un error común trabajar en la solución dentro del compactado, lo cual provoca que los cambios no se guarden. Si usted comete este error y entrega una solución vacía, no tendrá oportunidad de reclamar.

El pequeño pueblo de Namba se caracteriza por cada año definir y llevar a cabo un variado grupo de proyectos en beneficio de la sociedad. Como parte del plan de desarrollo de este año, su gobernador se comprometió a disminuir el índice de vandalismo, el cual había aumentado en los últimos meses. Con el objetivo de instalar cámaras de seguridad de alta tecnología para favorecer la vigilancia, el gobernador contrató a una empresa extranjera.

Estas cámaras que se desean instalar tienen una calidad tal, que al ubicar una de ellas en una de las esquinas del pueblo, se garantiza el monitoreo de todas las calles adyacentes esta, por lo que se piensa que no será necesario poner cámaras en todos los puntos de conexión (esquinas) del pueblo.

Por ejemplo, la **Figura 1**, muestra un conjunto de puntos de conexión (C_i) donde pudieran ser instaladas las cámaras y las calles que estos puntos conectan (a_i).

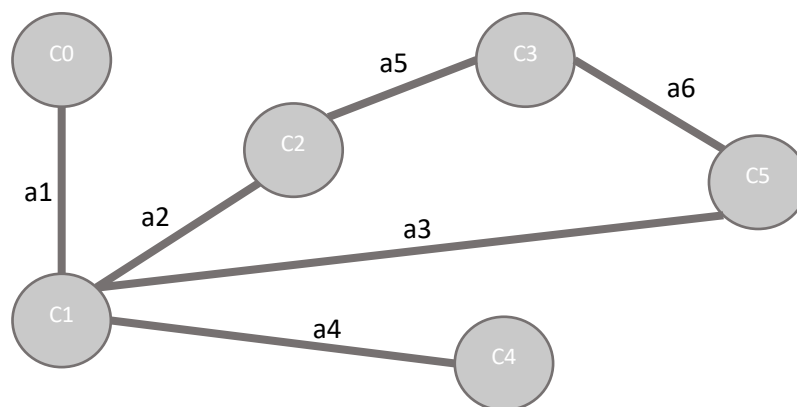


Fig 1 Mapa de las conexiones y calles de Namba.

En este caso, si se decidiera ubicar una cámara en el punto C_1 (**Figura 2**), entonces quedarían aseguradas las calles $\{a_1, a_2, a_3, a_4\}$,

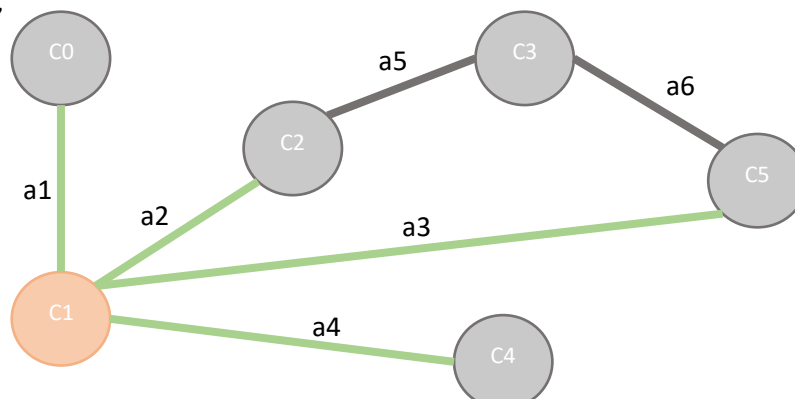


Fig 2 Instalación de una cámara en el punto de conexión C_1 .

Sin embargo, como te podrás imaginar, estas cámaras son muy caras y el presupuesto del proyecto no es infinito, por lo que el gobernador desea que lo ayudes a determinar **la menor cantidad de cámaras que debe instalar de manera tal que pueda monitorear todas las calles de la ciudad.**

Usted debe haber recibido junto a este documento una solución de C# con dos proyectos: una biblioteca de clases (*Class Library*) y una aplicación de consola (*Console Application*). Deberá implementar el método `AsignarCamaras` que se encuentra en la clase `Camaras` en el *namespace* `Weboo.Examen`. En la biblioteca de clases encontrará la siguiente definición:

```
namespace Weboo.Examen
{
    public class Camaras
    {
        public static int AsignarCamaras(bool[,] conexiones)
        {
            //Borre la siguiente línea y escriba su código
            throw new NotImplementedException();
        }
    }
}
```

El método recibirá como entrada:

- `bool[,] conexiones`: Representación computacional de los puntos de conexión y las calles que estos comunican. Que la posición `[i, j]` de esta matriz tenga valor `True`, significa que existe una calle entre los puntos C_i y C_j .

El método deberá retornar:

- `int`: La menor cantidad de cámaras que se necesitan instalar de manera tal que se logre monitorear todas las calles de la ciudad.

A continuación, se muestra cómo quedarían representados los datos del ejemplo inicial:

Entrada:

	C0	C1	C2	C3	C4	C5
C0	False	True	False	False	False	False
C1	True	False	True	False	True	True
C2	False	True	False	True	False	False
C3	False	False	True	False	False	True
C4	False	True	False	False	False	False
C5	False	True	False	True	False	False

`bool[,] conexiones`

Salida: **2** (en este caso con solo ubicar cámaras en los puntos C_1 y C_3 , se logran asegurar todas las calles tal como muestra la **Figura 3**).

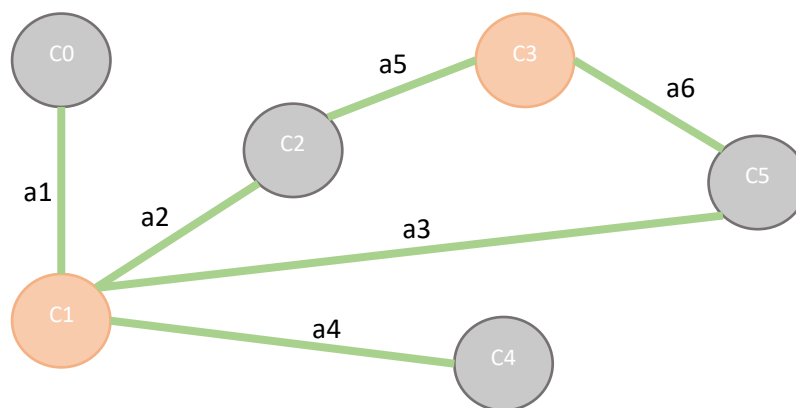


Fig 3 Instalación de cámaras en los puntos C1 y C3.

Nota que:

- En caso de que no existan calles entre ninguno de los puntos de conexión, entonces no será necesario instalar ninguna cámara.
- No siempre será útil ubicar una cámara en el punto con mayor cantidad de conexiones, como muestra la **Figura 4**.

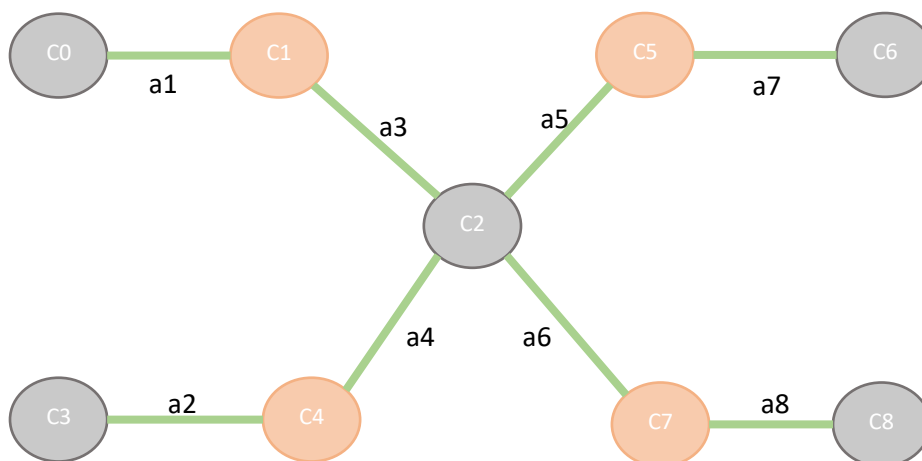


Fig 4 Nueva configuración de puntos de conexión y calles.

En este caso con solo 4 cámaras (ubicadas en los puntos C_1 , C_4 , C_5 y C_7) se logran monitorear todas las calles.

NOTA: Todo el código de la solución debe estar en este proyecto (biblioteca de clases), pues es el único código que será evaluado. Usted puede adicionar todo el código que considere necesario, pero no puede cambiar los nombres del namespace, clase o método mostrados. De lo contrario, el probador automático fallará. En particular, es imprescindible que usted no cambie el parámetro del método `AsignarCamaras`. Por supuesto, usted puede (y debe) adicionar todo el código que necesite.

NOTA: Los casos de prueba que aparecen en este proyecto son solamente de ejemplo. Que usted obtenga resultados correctos con estos casos no es garantía de que su solución sea correcta y de buenos resultados con otros ejemplos. De modo que usted debe probar con todos los casos que considere convenientes para comprobar la validez de su implementación.