

Ordenando proyectos

Una solución en *Visual Studio* está formada por uno o más proyectos entre los cuales pueden establecerse relaciones de dependencia. Si un proyecto **A** depende de un proyecto **B**, entonces el proyecto **B** debe ser compilado antes que el **A**. Las dependencias entre proyectos pueden ocasionar que no exista ninguna ordenación de los mismos tal que todo proyecto se compile antes que los que dependen de él (por ejemplo, si el proyecto **A** depende del **B** y el **B** depende del **A**).

Implemente una biblioteca de clases de nombre `OrdenaProyectos.DLL` (le recomendamos nombrar el proyecto `OrdenaProyectos`, sin la terminación `.DLL`) con la siguiente clase:

```
namespace OrdenaProyectos
{
    public class Problema3
    {
        public static int[] OrdenDeCompilacion(bool[,] dependencias)
        {
            // aquí va su implementación
        }
    }
}
```

El método `OrdenDeCompilacion` debe devolver una ordenación de un conjunto de proyectos acorde con sus dependencias. Los proyectos se identificarán a través de valores enteros empezando en 0. Las dependencias entre los proyectos se suministran mediante el *array* bidimensional `dependencias`. Un valor de `true` en `dependencias[i, j]` indica que el proyecto `i` depende del `j`. Se garantiza que `dependencias != null`

y que

`dependencias.GetLength(0) == dependencias.GetLength(1)` (es decir la matriz es cuadrada) siendo la longitud de las filas y las columnas la cantidad de proyectos.

La ordenación devuelta debe ser un *array* de longitud igual a la cantidad de proyectos y que cuyos valores nos den una permutación de los números de los proyectos. Note que puede haber más de una ordenación válida. En caso de no existir ninguna ordenación satisfactoria, el método `OrdenDeCompilacion` debe devolver `null`.

Dependencias	Ejemplos de respuestas correctas
<pre>{ {false, false, false, false}, {false, false, false, false}, {false, false, false, false}, {false, false, false, false}, }</pre> (no existen dependencias)	En este caso como no hay dependencias cualquier permutación es correcta

Dependencias	Ejemplos de respuestas correctas
<pre>{ {false, false, true, false}, {false, false, false, false}, {false, false, false, false}, {false, true, false, false}, }</pre> <ul style="list-style-type: none"> • 0 depende de 2 • 3 depende de 1 	<pre>{ 2, 0, 1, 3 } { 1, 2, 3, 0 }</pre>
<pre>{ {false, false, true, true}, {false, false, false, false}, {false, true, false, true}, {false, true, false, false}, }</pre> <ul style="list-style-type: none"> • 0 depende de 2 y 3 • 2 depende de 1 y 3 • 3 depende de 1 	<pre>{ 1, 3, 2, 0 }</pre> (única respuesta correcta)
<pre>{ {false, true, false, false}, {false, false, true, false}, {true, false, false, false}, {false, false, false, false}, }</pre> <ul style="list-style-type: none"> • 0 depende de 1 • 1 depende de 2 • 2 depende de 0 	null (no hay ninguna ordenación posible)

Aclaraciones

- Un proyecto nunca dependerá de sí mismo, es decir, `dependencias[i, i] == false` para todo `i`.
- Al menos debe probar con casos de cantidad de proyectos menor o igual a 10.