

Programa: Ingeniería en Computación  
Curso: IC-5701 Compiladores e intérpretes (2018-2)  
Proyectos: **Manual para integración del IDE y el compilador de  $\Delta$**   
Autor: Diego Ramírez Rodríguez

## IDE-Triangle – Integración con el compilador de $\Delta$

Diego Ramírez Rodríguez  
Ingeniería en Computación, Tecnológico de Costa Rica

---

### Presentación

En este documento complementamos lo descrito en el documento '**ide-triangle.pdf**', que está incluido en la carpeta 'Triangle Java IDE LL Pérez + Navarro & Fallas.zip'.

### Guía

**\*Se recomienda utilizar NetBeans IDE 8.2 para realizar el trabajo\***

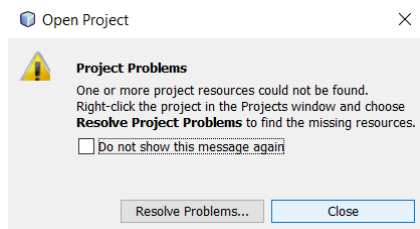
Se necesitan los archivos:

1. "Triangle\_Java\_IDE\_LL\_Pérez\_\_\_\_Navarro\_\_\_\_Fallas.zip"
2. "TriangleJava.zip"

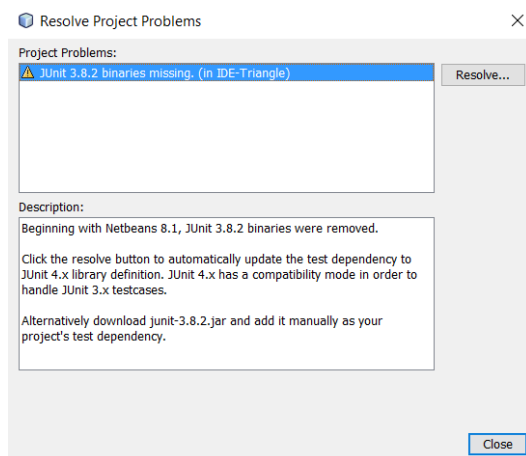
Descomprima ambos.

El proyecto de NetBeans sobre el que se trabajará es: "ide-triangle-v1.1.src".

Abra este archivo desde NetBeans. De entrada, aparece una ventana similar a esta:

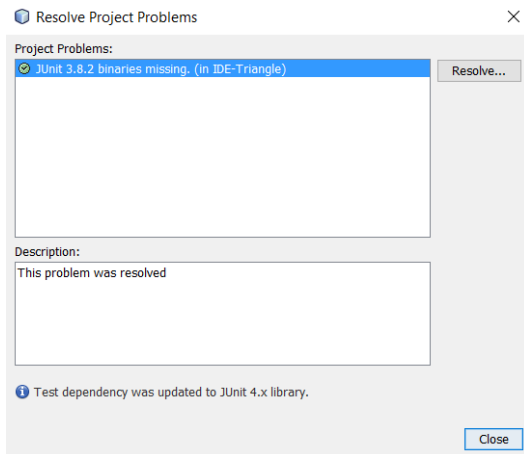


Seleccione "Resolve Problems..." y se le presentará lo siguiente:

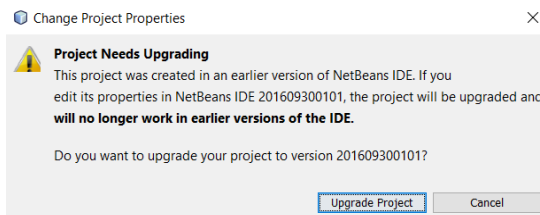


Programa: Ingeniería en Computación  
Curso: IC-5701 Compiladores e intérpretes (2018-2)  
Proyectos: **Manual para integración del IDE y el compilador de  $\Delta$**   
Autor: Diego Ramírez Rodríguez

Seleccione “Resolve...” y cierre la ventana una vez que se actualice así:



En el apartado “Libraries” elimine “Triangle.jar” y debería aparecer una ventana similar a esta:



Seleccione “Upgrade Project”.

Cierre el proyecto en NetBeans.

En el explorador de archivos entre a "ide-triangle-v1.1.src", y elimine “Triangle.jar”.

Luego entre a la carpeta “src”.

Por aparte, abra la carpeta “TriangleJava” que descomprimió antes desde “TriangleJava.zip” y siga estos pasos:

TriangleJava > Triangle-tools-2.1.zip > tools-2.1

Mueva la carpeta “TAM” de “tools-2.1” a “src”.

Mueva el contenido de la carpeta “Triangle” en “tools-2.1” adentro de “Triangle” en “src”.

El proyecto ya está funcional.

---

1. Desactivar/cambiar las siguientes líneas dentro de Main.java dentro del código del IDE:

- **Línea 619 (comentar):**  
disassembler.Disassemble(desktopPane.getSelectedFrame().getTitle().replace(".tri", ".tam"));

Programa: Ingeniería en Computación  
Curso: IC-5701 Compiladores e intérpretes (2018-2)  
Proyectos: **Manual para integración del IDE y el compilador de  $\Delta$**   
Autor: Diego Ramírez Rodríguez

- **Línea 621 (comentar):**  
`((FileFrame)desktopPane.getSelectedFrame()).setTable(tableVisitor.getTable(compiler.getAST()));`
- **Línea 623 (cambiar de true a false):**  
`runMenuItem.setEnabled(true);`
- **Línea 624 (cambiar de true a false):**  
`buttonRun.setEnabled(true);`

2. **IDECompiler.java:** en realidad, el IDE nunca llama al `Compiler.java` del paquete de Triangle. Se crea uno propio llamado `IDECompiler.java`. Ahí llama al analizador contextual (`Checker`) y al generador de código (`Encoder`). Desactívelos cambiando las siguientes líneas:

- **Líneas 56,57,58 (comentar):**  
`System.out.println("Contextual Analysis ...");`  
`Checker checker = new Checker(report);`  
`checker.check(rootAST);`
- **Líneas 60,61,62 (comentar):**  
`System.out.println("Code Generation ...");`  
`Encoder encoder = new Encoder(report);`  
`encoder.encodeRun(rootAST, false);`
- **Línea 65 (comentar):**  
`encoder.saveObjectProgram(sourceName.replace(".tri", ".tam"));`