Universidad de Murcia

Facultad de Informática

Manual de Uso del Ejecutable

Sistemas Inteligentes 2024/2025

Índice

1. Introducción

Este programa simula un motor de inferencia con encaminamiento hacia atrás, utilizando un conjunto de hechos y reglas para alcanzar un objetivo definido. A través de un sistema basado en reglas, se verifica si un objetivo puede ser satisfecho. Por último, genera un archivo de texto con lo resultados, para cada base de hechos procesada.

2. Requisitos

- **Lenguaje de Programación:** C++
- **Compilador:** GCC o cualquier compilador C++17 o superior
- **Archivos de Entrada:**
  - Base de Conocimientos (un archivo con reglas)
  - Bases de Hechos (uno o varios archivos con hechos)

3. Estructura del programa

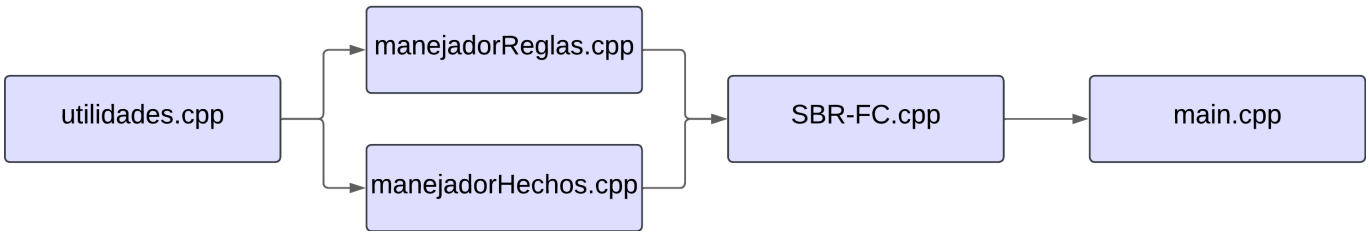


Figura 1: Diagrama módulos del programa

El programa se creó modularizado, con tal de facilitar la programación y evitar un código bastante engorroso. Pues, si metes todas las funciones en un único fichero, queda bastante largo y es más difícil de comprender. En el README.md se explica cada fichero, y lo que implementa. Adicionalmente, en los archivos del código se introducen comentarios.

3.1. Compilación

El programa fue compilado utilizando el compilador GCC versión 13.2.0, que también es compatible con el estándar C++17, y con versiones recientes como C++20 y C++23.

En vscode, se configuró un archivo "tasks.json" (dentro de .vscode) para automatizar la compilación del proyecto completo utilizando el comando **Ctrl+Shift+B**. Este archivo permite compilar todas las dependencias automáticamente y generar el ejecutable fácilmente.

No obstante, en caso de que no funcione, siempre se puede realizar la compilación manual, desde la terminal. Se emplearían los siguientes comandos:

```
g++ -c utilidades.cpp -o utilidades.o
g++ -c manejadorHechos.cpp -o manejadorHechos.o
g++ -c manejadorReglas.cpp -o manejadorReglas.o
g++ -c SBR-FC.cpp -o SBR-FC.o
g++ -c main.cpp -o main.o
g++ main.o manejadorHechos.o manejadorReglas.o SBR-FC.o utilidades.o -o programa
```

4. Ejecución del programa

El programa se ejecuta desde la línea de comandos, como cmd, de la siguiente manera:

```
$ .\programa.exe <ruta_BC> <ruta_BH1> [<ruta_BH2> ...]
```

- `ruta_BC` : Ruta del archivo que contiene la base de conocimiento.
- `ruta_BH1 , ruta_BH2 , ...`: Rutas de los archivos que contienen bases de hechos. Puede ser un fichero, o varios.

Ejemplo:

```
$ .\programa.exe BC-1.txt BH-1.txt
$ .\programa.exe BC-2.txt BH-2-EST.txt BH-2-RM.txt
```

4.1. Estructura parámetros de entrada

4.1.1. Base de Conocimientos (BC)

El archivo debe comenzar con el número total de reglas y seguir con las reglas en el siguiente formato:

```
<numero_de_reglas>
R<número>: Si <antecedentes> Entonces <consecuente>, FC=<factor_de_certeza>
```

- **Antecedentes:** Pueden ir separados por `y` o `o`.

4.1.2. Base de Hechos (BH)

El archivo debe comenzar con el número de hechos, el listado de los hechos y el objetivo final:

```
<numero_de_hechos>
<nombre_hecho>, FC=<factor_de_certeza>
Objetivo
<objetivo>
```

4.2. Estructura salida del programa

El programa genera un archivo de texto, con los resultados para cada base de hechos procesada. El archivo se guarda con el nombre:

```
Resultado_<nombre_base_hechos>.txt
```

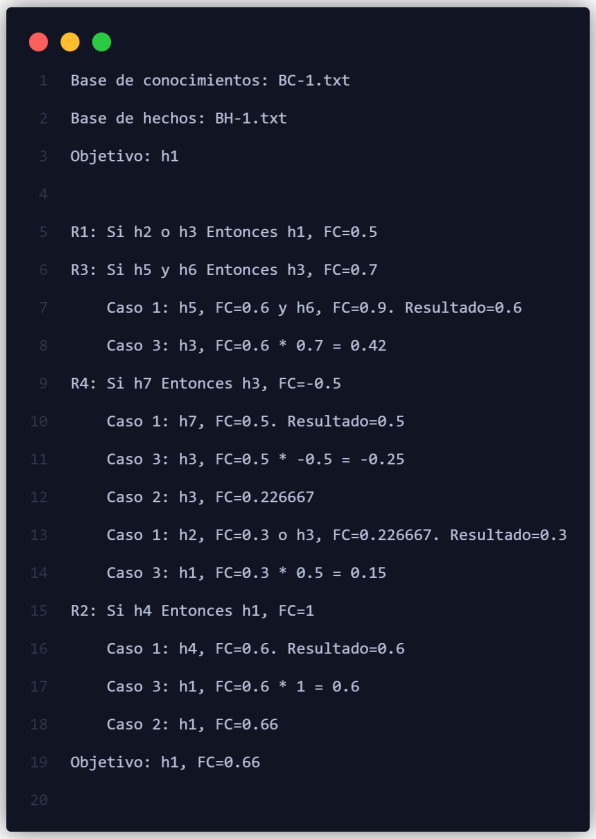
Este archivo contiene:

- Las reglas aplicadas
- Los factores de certeza calculados
- El resultado final del proceso de encaminamiento hacia atrás

Para la práctica, hemos obtenido los siguientes ficheros de resultados:

- `Resultado_BC-1_BH-1.txt`
- `Resultado_BC-2_BH-2-EST.txt`
- `Resultado_BC-2_BH-2-RM.txt`
- `Resultado_BC-3_BH-3.txt`
- `Resultado_BC-A_BH-A.txt`

Ejemplo fichero de salida



En la figura [2], podemos ver un ejemplo del fichero de salida para la prueba 1, que incluye los siguientes puntos:

- **Nombres de las bases de conocimiento (BC) y hechos (BH):** Aparecen al principio del fichero.
- **Objetivo especificado**: Se indica al principio el objetivo que tenemos, y al final, se vuelve a indicar con el resultado final del encaminamiento.
- **Proceso de inferencia:** Se indican los casos realizados para alcanzar el objetivo, y mostrando cada regla empleada.

Esto asegura que cumple el formato pedido en la práctica.

Figura 2: Ejemplo salida para la prueba 1.

5. Bibliografía

Vídeo para compilar varios archivos en vscode: `https://www.youtube.com/watch?v=q0YKEuFoxhQ&ab_channel=ChristianVargas`