本项目遵循以下代码风格规范，以保持一致性、可读性和可维护性：

**文件组织**

**/工程目录**

**│── .vscode/ #** (gdb不User Friendly，**.vscode下各个文件（settings.json、tasks.json 等）用以配置**“有GUI的gdb环境”。**（settings.json、tasks.json 等）**均由gpt生成

**│── doc/ # 文档目录（通常用于存放项目相关的文档）**

**│── include/ # 头文件目录（存放 .h头文件）**

**│── out/ # 编译输出目录（存放 .o目标文件）**

**│── pic/ # 图片资源目录**

**│── src/ # 源代码目录（存放 .cpp文件）**

**│── makefile # Makefile 编译脚本（用于自动化构建）**

**头文件** .h：仅包含 class 声明（和一部分内敛(inline)函数），使用 #ifndef 避免重复包含。

**源文件** .cpp：包含 class 的具体定义。

.h 和 .cpp 文件命名与 class 名保持一致，例如 S\_Ida\_CXL\_Type.h 和 S\_Ida\_CXL\_Type.cpp。

**类成员类型限定**：

仅使用基本数据类型

bool(1bit)

unsigned int(32 bit), 里面只用u32

unsigned long long(64 bit)，里面只用u64

U512

使用

Std::string

仅使用标准库容器，如 std::list<T>。std::vector, std::unordered\_map

仅使用本项目定义的其他 class。

不定义新的数据结构，如 struct 或 typedef。

**函数参数约定**：

**输入参数**：统一使用 const T&，避免不必要的拷贝，提高效率。

**输出参数**：统一使用 T&，保证数据修改可见，避免返回值拷贝。

**返回值类型**：为基本数据类型，或者void

**类的组织与初始化**：

类的构造函数应初始化所有成员变量，避免未定义行为。

只要实现类的构造函数，用CLASS (T a) : a(0),这里的T只是基本数据类型，其他数据类型不用特意初始化

不要实现类的构造函数

不使用继承的

禁止使用动态内存分配（new，delete），所有数据均应为自动管理的栈对象或标准容器。

不使用 friend 关键字，避免破坏封装性。

    std::list<S2M\_Snp\_type> s2m\_snp\_dpa;

    std::list<S2M\_NDR\_Type> s2m\_ndr\_dpa;

    std::list<S2M\_DRS\_Type> s2m\_drs\_dpa;

    main\_product(m2s\_rsp\_dpa, m2s\_rwd\_dpa, s2m\_snp\_dpa, s2m\_ndr\_dpa, s2m\_drs\_dpa, rd\_data, rd\_addr, wr);

之后，要写出s2m\_snp\_dpa, s2m\_ndr\_dpa, s2m\_drs\_dpa = main\_product(m2s\_rsp\_dpa, m2s\_rwd\_dpa)

其他的方式不要定义push，也不要出现[]

容器成员的大小要被控制，当push（list）出现和 []（map）出现时，必须检查size，防止栈/堆内存溢出。

**类命名规范**

所有类名遵循 S\_Ida\_CXL\_Type 形式。

单词间使用 \_ 分割，首字母大写，保持统一风格。

**变量命名规范**

变量命名格式为 s\_ds\_dsaf\_ds。

单词间使用 \_ 分割，全部小写。

避免使用缩写，确保变量名清晰易读。

**访问控制**

仅使用 private 和 public，不使用 protected。

成员变量必须设为 private，对外提供 public 方法访问。

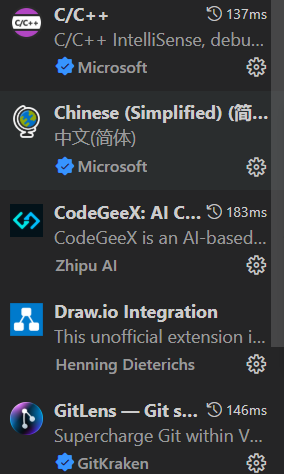
**面向对象特性**

**不使用继承和多态**，所有 class 独立实现。

通过组合方式（成员变量为其他 class）实现功能扩展。

环境 gcc12.2.0 , make3.81+, 兼容Linux，windows。要设为环境变量

代码编写环境windows下：vscode+如下插件



Makefile , 和.vscode(vscode下user-friendly的gdb环境)均由gpt生成