**各类公共数据结构设计**

**＊统一将类分析表等作为类，初始化时实例化，向外以API提供交互**

**符号表SymbolList**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name: string | Type: int(类型表偏移量) | Category: enum Cat(编码) | Offset: int(根据不同cat表示各类表偏移量) | Activity:bool(变量活跃信息,只有变量有意义,默认为true) |

API:

|  |  |
| --- | --- |
| setName(int position, string name) | 设置Name项 |
| setType(int position, int type) | 设置Type项 |
| setCategory(int position, Cat category) | 设置Category项 |
| setOffset(int position, int address) | 设置Offset项 |
| setActivity(int position, bool activity) | 设置Activity项 |
| find(string name) | 查找是否存在name，不存在返回-1，否则返回符号表中位置 |
| insert(string name) | 插入name新项并返回新项在符号表中的位置 |
| getName(int position) | 返回position位置的name,不存在则返回空串 |
| getType(int position) | 返回position位置的Type值 |
| getCategory(int position) | 返回position位置的Category值 |
| getActivity(int position) | 返回position位置的Activity值 |

**Token结构**

|  |  |  |  |
| --- | --- | --- | --- |
| Category:union Cat(编码) | Offset:int (偏移量) | Activity:bool(活跃信息) | Position:int |

API:

|  |  |
| --- | --- |
| setCategory(Cat category) | 设置Category值 |
| setOffset(int offset) | 设置Offset值 |
| setActivity(bool activity) | 设置活跃信息(默认不设置初始化为true) |
| setPosition(int position) | 设置position值,用于数组结构体的成员变量寻址 |
| getCategory() | 返回Category值 |
| getOffset() | 返回Offset值 |
| getActivity() | 返回活跃信息 |
| getPosition() | 返回position值 |

**类型表TypeList**

|  |  |
| --- | --- |
| Type: enum TypeValue(编码) | Offset: int(同符号表Offset一致) |

API:

|  |  |
| --- | --- |
| setType(int position, TypeValue type) | 设置Type值 |
| setOffset(int position, int offset) | 设置Offset值 |
| getType(int position) | 返回Type值 |
| insert(int TypeValue type, int offset) | 插入自定义类型 |

**各类常量表Template**

|  |
| --- |
| Value: various type |

API:

|  |  |
| --- | --- |
| set(int position, <T> value) | 设置值 |
| get(int position) | 返回值 |
| insert(<T> value) | 插入value并返回position |
| size() | 返回表大小 |
| find(<T> value) | 查找表中是否存在value,存在则返回offset,否则返回-1 |

**四元式格式Quater**

|  |  |  |  |
| --- | --- | --- | --- |
| 操作option | 第一操作数first | 第二操作数second | 目标target |

四元式里面操作数和目标均为token信息

option: std::string {+,-,\*,/, =, ==, >, <, >=, <=, if, ie, else, do…}

API:

|  |  |
| --- | --- |
| setOption(std::string option) | 设置option的值 |
| setFirst(Token token) | ... |
| setSecond(Token token) | ... |
| setTarget(Token token) | ... |
| getOption() | 返回option的值 |
| getFirst() | ... |
| getSecond() | ... |
| getTarget() | ... |

**数组信息表ArrayInformation**

|  |  |  |
| --- | --- | --- |
| int length(数组长度) | int type(成员类型,指向typeList) | int size(成员大小) |

API:

|  |  |
| --- | --- |
| getLength(int position) | 获取第position项length |
| getSize(int position) | ...size |
| getType(int position) | ...type |
| setLength(int position) | 设置第position项的length |
| setType(int position) | ...type |
| setSize(int position) | ...size |
| insert(int length, int type, int size) | 插入一条数组信息,并返回该信息在数组信息表中的位置 |