

Идиома RAII — конспект темы

Жизненный цикл объекта

Вся программа разделена на блоки, ограниченные фигурными скобками:

```
int Function () {  
    // Начало блока 1  
    ...  
    while (condition) {  
        // Начало блока 2  
        if (condition) {  
            // Начало блока 3  
            ...  
        } // Конец блока 3  
        ...  
    } // Конец блока 2  
    ...  
} // Конец блока 1
```

Объекты, объявленные внутри блока **без использования динамической памяти**, считаются автоматическими и «живут» на стеке. При создании объектов вызывается их конструктор. В конце блока все автоматические объекты разрушаются. Вызываются их деструкторы.

Если объекты создаются **в динамической памяти**, их жизненным циклом начинает управлять программист.

Если внутри блока выброшено исключение, все деструкторы автоматических объектов, созданных до этого, будут вызваны в обратном порядке — это **раскрутка стека**.

Раскрутка будет продолжаться до тех пор, пока кто-нибудь не поймает исключение или пока программа не закончит работу аварийно.

Деструкторы должны гарантировать отсутствие исключений.

Идиома RAII (Resource acquisition is initialization) заключается в том, что инициализация объекта должна быть совмещена с началом жизни объекта, а освобождение ресурсов, используемых этим объектом, должно происходить при уничтожении объекта.

Идея RAI

Ресурс — что-либо выданное объекту во временное пользование третьей стороной. Например, мьютексы, дескрипторы файлов или память.

Инициализация — создание некой автоматической переменной, которая сразу инициализируется. В конструкторе автоматическая переменная инициализируется и ресурс выделяется, а в деструкторе ресурс освобождается.

RAI в стандартных библиотеках

Примеры реализации RAI в стандартных библиотеках:

- **Контейнеры.** Самый распространённый пример RAI — почти любой контейнер кроме `array`. В `array` нет указателя на динамическую память.
- **Умные указатели.**
- **Мьютексы.**
- **Файлы.** Классы для работы с файлами в C++ — уже сами по себе RAI-обёртки над более низкоуровневыми функциями работы с файловой системой. Файл открывается в конструкторе и закрывается в деструкторе.