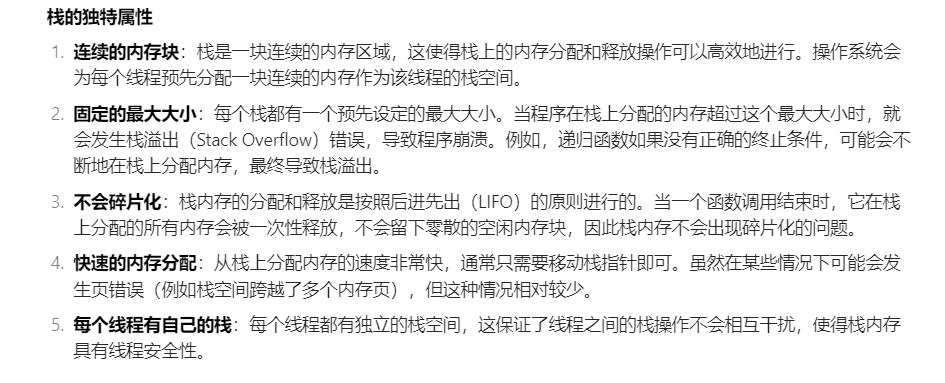
**栈的特点**

* **存储内容**：栈主要用于存储局部变量和函数参数。当函数被调用时，会在栈上为函数的局部变量和参数分配内存空间；当函数返回时，这些内存空间会被自动释放。
* **增长与收缩**：每次调用函数时，栈指针会向下移动，为新的局部变量和参数分配空间，栈的大小会增长；当函数返回时，栈指针会向上移动，释放之前分配的空间，栈的大小会收缩。
* **线程安全性**：每个线程都有自己独立的栈，不同线程的栈是相互隔离的，因此栈内存可以被认为是线程安全的，一个线程的栈操作不会影响其他线程的栈。

**堆的特点**

* **共享性**：堆是一个全局的内存区域，在同一个进程的所有线程之间共享。多个线程可以同时访问和操作堆中的内存。
* **动态分配与释放**：在 C++ 中，我们使用 new 运算符（或 C 库函数 malloc() 和 calloc()）在堆上动态分配内存，此时堆的大小会增长；使用 delete 运算符（或 free() 函数）释放内存时，堆的大小会收缩。
* **增长方向**：通常情况下，堆从低地址开始向上增长，随着内存的不断分配，堆的地址会逐渐升高。



图形用户界面, 文本, 应用程序, 电子邮件

AI 生成的内容可能不正确。

栈内存的分配和释放模式是顺序的，也就是说，内存总是按照与分配时相反的顺序释放。文本

AI 生成的内容可能不正确。