**C++ 技术管理文档**

**等级说明：**

* **a. 强制**：必须遵守，违反将导致代码审核不通过。
* **b. 推荐**：最佳实践，非强制但需充分理由才能偏离。
* **c. 允许**：在特定场景下可使用，但需谨慎。

**1. 头文件与包含**

* a1. 头文件必须使用 **#pragma once** 或标准 **#define** 宏守卫防止重复包含。
* a2. 禁止在头文件中使用 **using namespace**（避免命名空间污染）。
* b1. 头文件应自包含（不依赖其他头文件的隐式包含）。
* c1. 允许在 **.cpp** 文件中使用 **using namespace std**（限作用域内）。

**2. 命名规范**

* a3. 变量/函数名使用 **snake\_case**，类/结构体使用 **PascalCase**，常量使用 **kCamelCase** 或 **UPPER\_CASE**。
* a4. 禁止使用单字符变量名（除循环计数器 **i, j, k**）。
* b2. 推荐用有意义的名称（如 **CalculateScore()** 而非 **Calc()**）。
* c2. 允许缩写（如 **num** 代替 **number**），但需团队统一。

**3. 类与对象**

* a5. 类成员变量必须为 **private**（通过接口访问）。
* a6. 单参数构造函数必须用 **explicit** 标记（避免隐式转换）。
* b3. 推荐使用 **override** 关键字标记虚函数重写。
* c3. 允许使用默认析构函数（若无需特殊逻辑）。

**4. 内存管理**

* a7. **禁止裸指针管理内存**（必须用 **std::unique\_ptr**/**std::shared\_ptr**）。
* a8. 动态数组必须用 **std::vector** 或 **std::array**（禁用 **new T[]**）。
* b4. 推荐用 **make\_unique**/**make\_shared** 替代直接 **new**。
* c4. 允许自定义内存池（需附性能测试报告）。

**5. 错误处理**

* a9. 禁止使用 C 风格错误码（改用异常或 **std::optional**/**std::expected**）。
* a10. 异常仅用于错误处理（**禁用**异常控制流程）。
* b5. 推荐用 **noexcept** 标记不抛异常的函数。
* c5. 允许在性能关键代码禁用异常（需全局统一编译选项）。

**6. 现代 C++ 特性**

* a11. 用 **nullptr** 替代 **NULL** 或 **0**。
* a12. 用 **auto** 简化迭代器类型（如 **auto it = vec.begin()**）。
* b6. 推荐用 **constexpr** 实现编译时计算。
* c6. 允许使用 Lambda 表达式（避免过度嵌套）。

**7. 格式化与可读性**

* a13. 缩进必须为 **4 个空格**（禁止 Tab）。
* a14. 单行长度不超过 **100 字符**（Google 标准为 80，放宽至 100）。
* b7. 推荐大括号 **{}** 统一换行（K&R 风格：**if (...) {**）。
* c7. 允许链式调用忽略换行（如 **obj.func1().func2()**）。

**8. 性能优化**

* a15. 禁止在循环中调用高开销函数（如 **strlen()**）。
* b8. 推荐用 **reserve()** 预分配容器内存。
* b9. 推荐用移动语义（**std::move**）优化拷贝。
* c8. 允许内联汇编（需附性能增益证明）。

**9. 模板与泛型**

* a16. 模板类型参数用 **typename** 而非 **class**（除非需兼容旧代码）。
* b10. 推荐用 **concepts**（C++20）约束模板参数。
* c9. 允许模板元编程（限基础库开发）。

**10. 多线程**

* a17. 共享数据必须用 **std::mutex** 或原子操作保护。
* a18. 禁止直接调用 **std::thread**（使用 **std::async** 或线程池）。
* b11. 推荐用 **std::atomic** 替代 **volatile**。
* c10. 允许无锁编程（需团队专家评审）。

**11. 宏与预处理**

* a19. 禁用宏定义常量（用 **constexpr** 变量替代）。
* a20. 禁用宏定义函数（用内联函数或模板替代）。
* b12. 推荐用 **static\_assert** 替代运行时断言（编译期检查）。
* c11. 允许平台条件编译（如 **#ifdef \_WIN32**）。

**12. 测试与维护**

* a21. 公共接口必须含单元测试（覆盖率 ≥80%）。
* b13. 推荐用 **clang-format** 自动化代码格式化。
* b14. 推荐用 Doxygen 风格注释公共 API。
* c12. 允许 **TODO** 注释标记临时代码（需附 Jira 编号）。

**13. 编译与工具**

* a22. 编译必须零警告（**-Wall -Wextra -Werror**）。
* a23. 禁用 C 标准库函数（优先用 C++ 替代，如 **memcpy** → **std::copy**）。
* b15. 推荐启用静态分析（Clang-Tidy/SonarQube）。
* c13. 允许使用 **#pragma** 抑制特定警告（需写明原因）。

**总计规则：41 条**（强制 23 条，推荐 15 条，允许 13 条）。