

# C++语言

这是一份核心 C++ 语言构造的参考手册。

## 基本概念

注释  
ASCII 码表  
标点符号  
名字与标识符  
类型 - 基础类型  
对象 - 作用域 - 生存期  
定义与 ODR  
名字查找  
有限定 - 无限定 (实参依赖)  
“如同”规则  
未定义行为  
内存模型与数据竞争  
字符集与编码  
翻译阶段  
main 函数  
模块 (C++20)

## 关键词

## 预处理器

#if - #ifdef - #ifndef -  
#elif  
#elifdef -  
#elifndef(C++23)  
#define - # - ##  
#include - #pragma  
#line - #error - #warning  
(C++23)

## 表达式

值类别  
求值顺序与定序  
常量表达式  
运算符  
赋值 - 算术  
自增与自减  
逻辑 - 比较  
成员访问与间接  
调用、逗号、三元  
sizeof - alignof (C++11)  
new - delete - typeid  
代用表示  
运算符重载  
默认比较 (C++20)  
运算符优先级  
转换  
隐式 - 显式 - 用户定义  
一般算术转换

## 声明

命名空间声明  
命名空间别名  
引用 - 指针 - 数组  
结构化绑定 (C++17)  
枚举与枚举项  
存储期与链接  
翻译单元局部 (C++20)  
语言链接  
inline 说明符  
内联汇编  
const/volatile - constexpr  
(C++11)  
constexpr (C++20) - constexpr  
(C++20)  
decltype (C++11) - auto  
(C++11)  
typedef - 类型别名 (C++11)  
详述类型说明符  
属性 (C++11) - alignas (C++11)  
static\_assert (C++11)

## 初始化

默认初始化  
值初始化 (C++03)  
复制初始化  
直接初始化  
聚合初始化  
列表初始化 (C++11)  
引用初始化  
静态非局部初始化  
零 - 常量  
动态非局部初始化  
有序 - 无序  
复制消除

## 函数

函数声明  
默认实参  
变长实参  
Lambda 表达式 (C++11)  
实参依赖查找  
重载决议  
运算符重载  
重载集的地址  
协程 (C++20)

## 语句

## 类

类类型 - 联合类型  
注入类名  
数据成员 - 位域  
成员函数 - this 指针  
静态成员 - 嵌套类  
派生类 - using 声明  
空基类优化  
虚函数 - 抽象类  
override (C++11) - final  
(C++11)  
成员访问 - friend  
构造函数与成员初始化器列表  
默认构造函数 - 析构函数  
复制构造函数 - 复制赋值  
移动构造函数 (C++11)  
移动赋值 (C++11)  
转换构造函数 - explicit 说明符

## 模板

模板形参与实参  
类模板 - 函数模板  
变量模板 (C++14)  
类成员模板  
模板实参推导  
类模板实参推导 (C++17)  
显式特化 - 部分特化  
形参包 (C++11) - sizeof...  
(C++11)  
折叠表达式 (C++17)  
待决名 - SFINAE  
约束与概念 (C++20)  
requires 表达式 (C++20)

## 异常

throw 表达式  
try-catch 块  
函数 try 块  
noexcept 说明符 (C++11)  
noexcept 运算符 (C++11)  
动态异常说明 (C++17 前)

## 杂项

C++ 的历史  
扩充命名空间 std  
字母缩写

static\_cast -  
dynamic\_cast  
const\_cast  
reinterpret\_cast  
字面量 (转义序列)  
布尔 - 整数 - 浮点  
字符 - 字符串  
nullptr (C++11)  
用户定义 (C++11)

if - switch  
for - 范围 for (C++11)  
while - do-while  
continue - break - goto -  
return  
synchronized 与 atomic (TM TS)

## 惯用手法

资源获取即初始化  
三/五/零法则  
指向实现的指针  
零开销原则  
奇特重现模板模式