版本号：v1.0 阶段：初稿

# 数据库面试宝典

MRL Liu

（未经许可，不得传播）

2022年02月02日

本文以《现代操作系统（原书第3版）》、小林Coding《图解系统》为参考，结合Linux操作系统介绍操作系统领域面试相关的知识，主要分为内存管理、进程线程、文件系统三个部分，深入研究可参考其他资料。

系统级软件是数字世界的基础设施，C++被誉为系统级编程皇冠上的明珠。

现代C++版本一般指的是国际C++标准委员会发布的4个版本：C++11/14/17/20。

如今，计算机行业已经诞生了无数优秀的现代编程语言，但是现在C++却仍然具有强盛的生命力。显然，C++语言具备独特的优势：**和C兼容，可以和硬件设备密切交互，完成各种稀奇古怪的底层功能**。本系列【面试C++】文章主要目的是记录当下作者关心的C++面试重要知识，若对您有所帮助，欢迎评论支持。

构造函数和析构函数是C++的核心。

C++11和之前的C++86相比更像是一种新语言。

面向对象程序设计的四大特性：封装、抽象、继承和多态。

标准的C++由三个重要部分组成：

* 核心语言，提供了所有构件块，包括变量、数据类型和常量等。
* C++标准库，提供了大量的函数，用于操作文件和字符串等。
* 标准模板库（STL），提供了大量的方法，用于操作数据结构等。

# C++基础语法篇

## 一、变量

C++中有常量、数组、字符串、指针、引用、类、结构体、联合体、枚举等各种概念，但是它们**本质上都是变量**，是的，你没听错，**编程语言中的常量也是一种特殊的变量**。

变量是什么？**变量本质是程序可操作的内存空间的名称，每个变量概念都由type和value组成**，其中type就是其**数据类型**，其决定了数据在内存中的存储方式和运算方式，**常量本质是定义后不可更改value的变量**。所以准确理解了变量概念，就相当于拿到了C++中的各种变量类型的总把手。

变量的value没什么好说的，但是变量的type可是五花八门，不同的type在内存中的大小也不一样，这里先简单说一些基本变量类型、指针和引用。

### 基础变量

基础变量就是变量的type为编程语言内置数据类型的变量，这些类型很常见，但是我们要**记住每种变量所占的内存字节数**。

|  |  |  |
| --- | --- | --- |
| **关键字** | **数据类型** | **内存大小** |
| bool | 布尔型 | —（实际测试为1个字节） |
| void | 无类型 | — |
| int | 整型 | 4个字节 |
| float | 浮点型 | 4个字节 |
| double | 双浮点型 | 8个字节 |
| char | 字符型 | 1个字节 |
| wchar\_t | 宽字符型 | 2或4个字节 |

注意：每一种数据类型的内存大小与其系统位数有关，上表列出的是目前为主的64位系统，一字节为8位。上述的一些基本类型（char、int、double）可以使用一个或多个类型修饰符进行修饰：signed、unsigned、short和long。

### 字符串变量

这里着重介绍下C++中的字符串，C++中有两种表示字符串的形式：C风格的字符数组和C++引入的string类。string类本质上也是封装了C风格的字符数组，但是其也集成了许多字符串的常见操作，是C++推荐的表示字符串的方式。

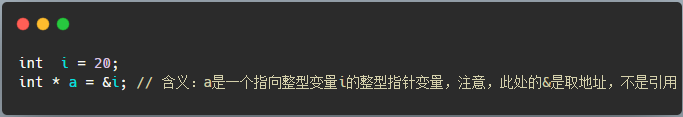
### 指针和引用变量

接下来我们介绍下**指针**和**引用**，这是面试的高频知识，这里我们只强调一些关键信息。

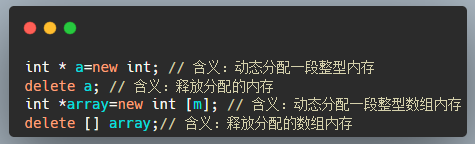
#### （1）指针

**指针本质上是一种特殊的变量，特殊在其value是另一个变量的地址，所以无论何种类型的指针，其内存大小都是固定的**（例如在X86中为4个字节，在X64中为8个字节）；**指针可以在任何时候初始化或者更改；存在空的指针**。

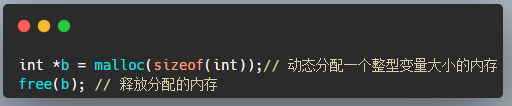
指针的定义如下：



但是实际编程中程序员不会这么无聊地用，基本都是new一个类对象，然后访问其属性和方法。new是一种操作符，和delete结合使用：



当然除了new和delete，也可以使用标准库函数malloc()和free()：



new和malloc()的主要区别：

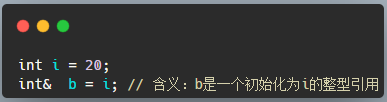
|  |  |  |
| --- | --- | --- |
|  | **new/delete** | **malloc()/free()** |
| **本质** | 运算符 | 标准库函数 |
| **分配大小** | 自动计算 | 作为参数手动输入 |
| **主要操作** | 分配/释放内存时调用对象的构造和析构函数 | 分配/释放内存时不调用对象的构造和析构函数 |
| **安全性** | **内存分配失败**时抛出bad\_alloc异常 | **内存分配失败**时返回null |
| **进行类型识别检测**，返回定义时具体类型指针，如果为int指针分配float变量时报错 | **不进行类型识别检测**，返回void指针，需要手动进行类型转换，如果为int指针分配float大小的字节数不报错。 |

总结，new封装了malloc()方法，new一般用来返回一个类对象，malloc(）一般用在基本变量（int、float），前者更加常用。

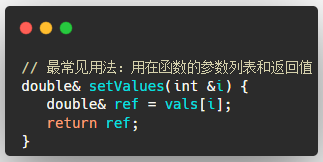
#### （2）引用

**引用**可能初学者不经常使用，但是没关系，**引用本质就是一个已存在的变量的别名，所以引用的内存分配区域和大小由其指向的变量类型决定**。需要注意的是**引用必须在创建时初始化，之后不可更改或再次初始化；不存在空的引用；这两点与指针很不同**。

引用的定义：



引用一般用在函数的参数和返回值上：



## 二、输入输出

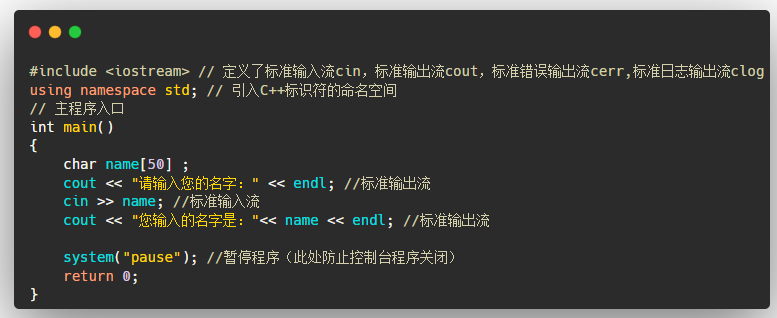
C++标准库提供了丰富的I/O（输入/输出）功能。首先看下C++编程中最基本和最常见的I/O操作。C++的I/O发生在流中，流是字节序列。

输入操作被C++看做字节流从设备（如键盘、磁盘驱动器、网络连接等）流向内存；

输出操作被C++看做字节流是从内存流向设备（如显示屏、打印机、磁盘驱动器、网络连接等）。

### 标准输入输出流

下面给出标准输入输出流的演示代码：



首先要用#include语句包含头文件<iostream>；using namespace std语句负责引入C++标识符的命名空间，没有该语句的话，则以下cout、cin和endl都得换成std::cout、std::cin和std::endl。cout可以直接输出字符串或字符数组，只在输出endl时换行；cin以阻塞的方式从控制台读取数据到字符数组中。

# C++标准库

本文先介绍C++的**编译流程**和**内存分区**，因为这是帮助我们理解

C++底层运行机制的基础知识，必须清楚；接着介绍C++中的**变量概念**，因为这是C++基础知识网络中的重要内容。

本篇文章我们希望你牢记：

（1）C++的编译流程，特别是预处理的位置。

（2）C++的内存分区，特别是堆和栈的特点。

（3）C++的变量概念，"一切皆变量"。

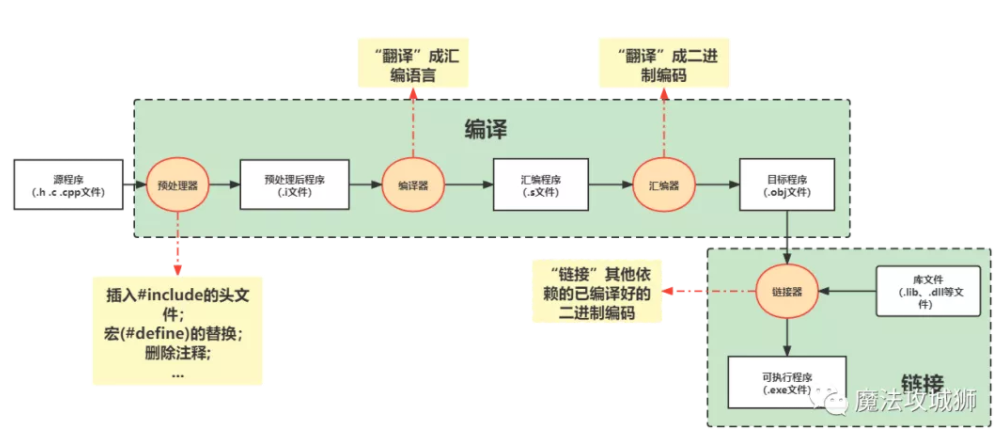
（4）new和malloc()的区别。

（5）指针和引用的区别。

## 一、C++的编译流程

我们写的C++源程序文件（.cpp）都要被IDE中集成的C++编译器（g++）编译生成.exe文件，实际上C++编译器做了两个步骤：**编译**和**链接**，其过程如下：

这个编译流程中需要注意的是预处理器的步骤，了解了这个步骤就能理解#define和#include，这张图的流程可能面试中不常问，但是最好做到心中有数。



## 二、C++的内存分区

了解了C++的编译流程，执行.exe文件，内存就会开始读入数据进行运算，此时我们就要知道C++在运行时各种数据的分配情况，这点在面试中也经常问到，也比较简单，熟记即可。

了解了C++的编译流程，执行.exe文件，内存就会开始读入数据进行运算，此时我们就要知道C++在运行时各种数据的分配情况，这点在面试中也经常问到，也比较简单，熟记即可。

# KV存储系统项目

使用跳表的数据结构

# 迭代日志表

|  |  |  |
| --- | --- | --- |
| **迭代版本** | **迭代工作** | **迭代日期** |
| V1.0 | 建立初稿文档，完成第1章和第2章的整理。 | 2022-02.02-02.05 |
| V1.1 |  |  |
| V1.2 |  |  |