

# Lesson00---前言

## 【本节目标】

- 1. 什么是C++
- 2. C++发展史
- 3. C++的重要性
- 4. 如何学习C++
- 5. 关于本门课程



## 1. 什么是C++

C语言是结构化和模块化的语言，适合处理较小规模的程序。对于复杂的问题，规模较大的程序，需要高度的抽象和建模时，C语言则不合适。为了解决软件危机，20世纪80年代，计算机界提出了OOP(object oriented programming: 面向对象)思想，支持面向对象的程序设计语言应运而生。

1982年，Bjarne Stroustrup博士在C语言的基础上引入并扩充了面向对象的概念，发明了一种新的程序语言。为了表达该语言与C语言的渊源关系，命名为C++。因此：C++是基于C语言而产生的，它既可以进行C语言的过程化程序设计，又可以进行以抽象数据类型为特点的基于对象的程序设计，还可以进行面向对象的程序设计。

## 2. C++的发展史

1979年，贝尔实验室的本贾尼等人试图分析unix内核的时候，试图将内核模块化，于是在C语言的基础上进行扩展，增加了类的机制，完成了一个可以运行的预处理程序，称之为C with classes。

语言的发展就像是练功打怪升级一样，也是逐步递进，由浅入深的过程。我们先来看下C++的历史版本。

阶段	内容
C with classes	类及派生类、公有和私有成员、类的构造和析构、友元、内联函数、赋值运算符重载等
C++1.0	添加虚函数概念，函数和运算符重载，引用、常量等
C++2.0	更加完善支持面向对象，新增保护成员、多重继承、对象的初始化、抽象类、静态成员以及const成员函数
C++3.0	进一步完善，引入模板，解决多重继承产生的二义性问题和相应构造和析构的处理
C++98	<b>C++标准第一个版本</b> ，绝大多数编译器都支持，得到了国际标准化组织(ISO)和美国标准化协会认可， <b>以模板方式重写C++标准库，引入了STL(标准模板库)</b>
C++03	C++标准第二个版本，语言特性无大改变，主要：修订错误、减少多异性
C++05	C++标准委员会发布了一份计数报告(Technical Report, TR1)，正式更名C++0x，即：计划在本世纪第一个10年的某个时间发布
C++11	增加了许多特性，使得C++更像一种新语言，比如：正则表达式、基于范围for循环、auto关键字、新容器、列表初始化、标准线程库等
C++14	对C++11的扩展，主要是修复C++11中漏洞以及改进，比如：泛型的lambda表达式，auto的返回值类型推导，二进制字面常量等
C++17	在C++11上做了一些小幅改进，增加了19个新特性，比如：static_assert()的文本信息可选，Fold表达式用于可变的模板，if和switch语句中的初始化器等
C++20	制定ing

C++还在不断的向后发展。

### 3. C++的重要性

#### 3.1 语言的使用广泛度

我们再看下今年的开发语言排行榜的情况：

Jun 2018	Jun 2017	Change	Programming Language	Ratings	Change
1	1		Java	15.368%	+0.88%
2	2		C	14.936%	+8.09%
3	3		C++	8.337%	+2.61%
4	4		Python	5.761%	+1.43%
5	5		C#	4.314%	+0.78%
6	6		Visual Basic .NET	3.762%	+0.65%
7	8	▲	PHP	2.881%	+0.11%
8	7	▼	JavaScript	2.495%	-0.53%
9	-	▲	SQL	2.339%	+2.34%
10	14	▲	R	1.452%	-0.70%

#### 3.2 在工作领域

1. 操作系统以及大型系统软件开发

2. 服务器端开发
3. 人工智能
4. 网络工具
5. 游戏开发
6. 嵌入式领域
7. 数字图像处理
8. 分布式应用
9. 移动设备

### 3.3 在校招领域

#### 3.3.1 岗位需求

## 腾讯2019校园招聘

注：毕业时间在2018年9月-2019年8月之间（国内院校以毕业证、海外院校以学位证时间为准）

软件开发	技术运营	安全技术	软件测试	技术研究
------	------	------	------	------

岗位方向

软件开发-后台开发方向

岗位描述

从事腾讯产品服务后台的架构设计、开发、优化及运营工作；使用最优秀的架构设计及算法实现，在网络接入、业务运行逻辑、用户数据存储、业务数据挖掘等方向，为海量互联网用户提供稳定、安全、高效和可靠的专业后台支撑体系。

岗位要求

编程基本功扎实，掌握C/C++/JAVA等开发语言、常用算法和数据结构；  
熟悉TCP/UDP网络协议及相关编程、进程间通讯编程；  
了解Python、Shell、Perl等脚本语言；  
了解MYSQL及SQL语言、编程，了解NoSQL, key-value存储原理；  
全面、扎实的软件知识结构，掌握操作系统、软件工程、设计模式、数据结构、数据库系统、网络安全等专业知识；  
了解分布式系统设计与开发、负载均衡技术，系统容灾设计，高可用系统等知识。

注：该岗位“招聘城市”在简历投递截止日前会有部分调整，请密切关注，腾讯公司对招聘信息保留最终解释权

### (CTO) 系统开发工程师

所属公司: 北京嘀嘀无限科...

所属部门: CTO线

工作地点: 北京市

招聘人数: 若干

学 历: 本科及以上

工作类型:

发布时间: 2018-09-03

#### 工作内容/职位描述:

1. 参与系统和网络相关产品开发
2. 参与系统和网络相关的监控和自动化运维平台的开发
3. 参与网络优化和系统优化的开发工作, 网络和系统前沿技术 (SDN、NFV等) 的研究和开发

#### 任职资格:

1. 良好的算法、数据结构基础, 熟练掌握如下语言中的一门: c/c++, python, java, shell, 并有一定开发经验
2. 精通TCP/IP协议和相关优化, 了解HTTP, SSL等协议
3. 熟悉linux操作系统, 有一定的linux系统使用和工作经验
4. 具有良好的敬业精神、团队合作能力和沟通能力

### 3.3.2 笔试题

[网易2018实习生招聘笔试题-C++开发实习生](#)

[2018迅雷校园招聘C++在线笔试A卷](#)

[爱奇艺2018秋季校招C++工程师 \(第一场\)](#)

### 3.3.3 面试题

#### 【某学长腾讯面试】

1. 讲解一下你的项目。【项目】
2. 讲一下epoll和select的区别。【网络】
3. epoll的ET和LT。【网络】
4. 红黑树的原理, 查询的时间复杂度。【数据结构/算法】
5. 进程间通信和线程间通信的方式和手段。【操作系统】
6. 共享内存的生命周期, 实现原理, 相关函数。【操作系统】
7. 你是怎么理解C++继承的。【C++】
8. 说一下多态。【C++】
9. 说一下纯虚函数, 抽象工厂模式是什么。【C++】 + 【设计模式】
10. 1万行中判断每行字符串出现的次数。【数据结构/算法】

#### 【某学长美团面经】

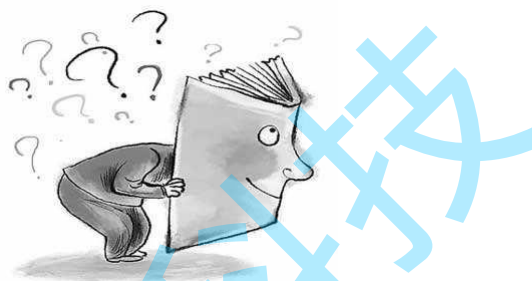
1. 进程地址空间? 【操作系统】
2. RAII/C++动态内存管理 【C++】
3. map底层, map底层为什么不用哈希表? C++中什么结构用了哈希表? 【数据结构/算法】
4. 线程池的设计? 【操作系统】
5. 如果是多计算型任务和多IO型任务, 在设计线程池时如何设计? 【操作系统】
6. 空间配置器? 【C++】

- 7. tcp相关, 就不列举了 【网络】
- 8. 实习期间的项目? 【项目】
- 9. map reduce使用场景? 【数据库】
- 10. 给你一个文件数据量很大, 怎么查找topK?(1.内存放不下 2.磁盘放不下) 【数据结构/算法】
- 11. weak\_ptr实现? auto\_ptr缺陷? 【C++】

从校招中公司岗位的技能要求, 以及学长面经总结了解到, 公司在校招期间更看重学生的基础, 最主要是: 语言(至少掌握一门面向对象语言java/C++)、数据结构、操作系统、网络、数据库、设计模式等, 而本门C++的授课内容, 更注重学生的实践动手能力、工作中的应用以及笔试面试中的技巧, 最后达到能够正常工作以及学习即可。

## 4. 如何学习C++

学习C语言需要多多练习, 学习数据结构需要多多思考, 那学习C++呢?



### 4.1 别人怎么学?

以下引用自2010年8月号《程序员》刊登的拙文《C++强大背后》最后一段:

C++缺点之一, 是相对许多语言复杂, 而且难学难精。许多人说学习C语言只需一本K&R《C程序设计语言》即可, 但C++书籍却是多不胜数。我是从C进入C++, 皆是靠阅读自学。在此分享一点学习心得。个人认为, 学习C++可分为4个层次:

- **第一个层次, C++基础** (平平常常)

挑选一本入门书籍, 如《C++ Primer》、《C++大学教程》或Stroustrup撰写的经典《C++程序设计语言》或他一年半前的新作《C++程序设计原理与实践》, 而一般C++课程也止于此, 另外《C++ 标准程序库》及《The C++ Standard Library Extensions》可供参考;

- **第二个层次, 正确高效的使用C++** (驾轻就熟)

此层次开始必须自修, 阅读过《(More)Effective C++》、《(More)Exceptional C++》、《Effective STL》及《C++编程规范》等, 才适宜踏入专业C++开发之路;

- **第三个层次, 深入解读C++** (出神入化)

关于全局问题可读《深入探索C++对象模型》、《Imperfect C++》、《C++沉思录》、《STL源码剖析》, 要挑战智商, 可看关于模版及模版元编程的书籍如《C++ Templates》、《C++设计新思维》、《C++模版元编程》;

- **第四个层次, 研究C++** (返璞归真)

阅读《C++语言的设计和演化》、《编程的本质》(含STL设计背后的数学根基)、C++标准文件《ISO/IEC 14882:2003》、C++标准委员会的提案书和报告书、关于C++的学术文献。由于我主要是应用C++，大约只停留于第二、三个层次。然而，C++只是软件开发的一环而已，单凭语言并不能应付业务和工程上的问题。建议读者不要强求几年内“彻底学会C++的知识”，到达第二层左右便从工作实战中汲取经验，有兴趣才慢慢继续学习更高层次的知识。虽然学习C++有难度，但也是相当有趣且有满足感的。

如果题主认为的「精通」是指上述所指的第二层次，那么我估计一年全职时间也未足够「精通」。首先，阅读这些书籍本身也需要不少时间，而且对于一般人来说（指除了一些学习记忆能力超强的人），很可能需要阅读几遍才能记住一些细节。另外，阅读后必须要练习。除了入门的习题外，还需要做一些工程向的项目，才能了解各种语法、编程范式的使用方法及时机，理想地建议参与一些开源项目。然而，学习是需要不断思考并把结果沉淀，方法包括与朋友讨论、写博客、写知乎答案等，这些都需要时间。

如引文末段所指出，我建议不要把「精通C++」作为一个一年目标，应该要把学习语言作为一个持续的过程，同时要把语言运用在具体的应用场合中。

<http://www.zhihu.com/question/23933514>

## 4.2 自己怎么学

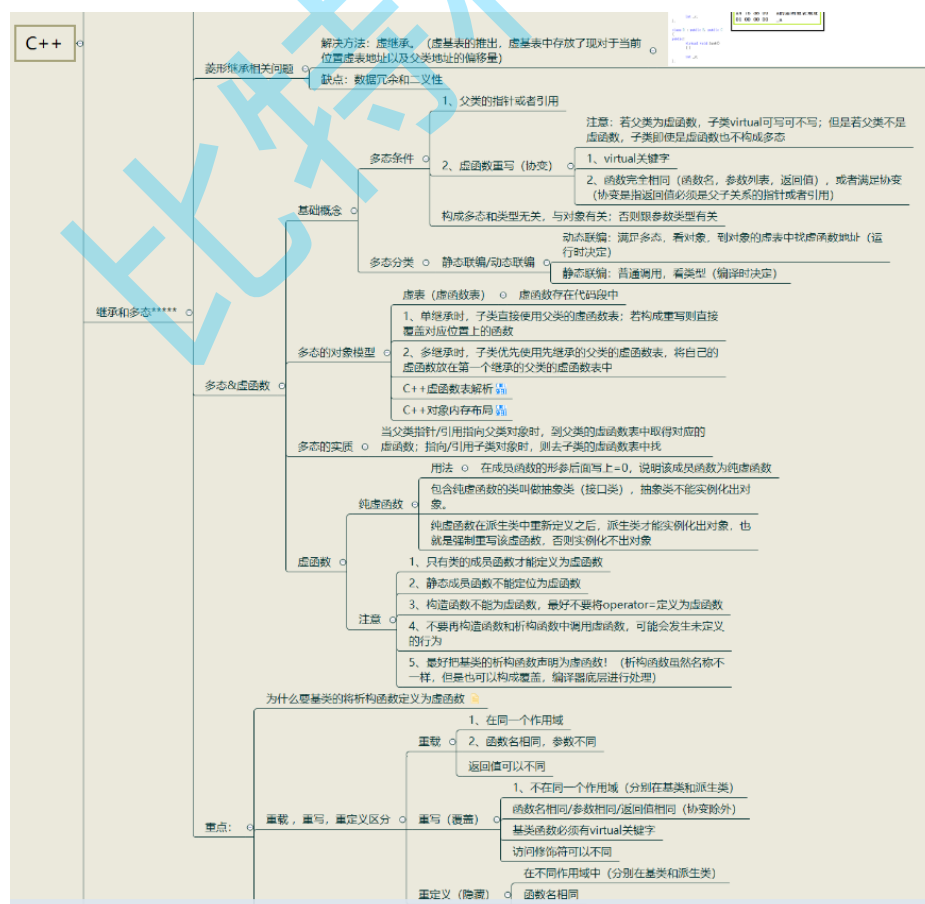
虽然C++语法特性繁琐复杂，经典书籍琳琅满目，但是只要你紧跟我们的节奏，你也可以“前途光明”。

### • 多总结

#### 1. 写博客

经常写博客不仅可以记录自己学习的历程，知识点的认识，更是复习的一把神兵利器。

#### 2. 画思维导图



画思维导图软件比如: xmind、imindmap等。



## • 常看书

合理的看书安排，切记贪，缓缓图之，否则容易走火入魔。

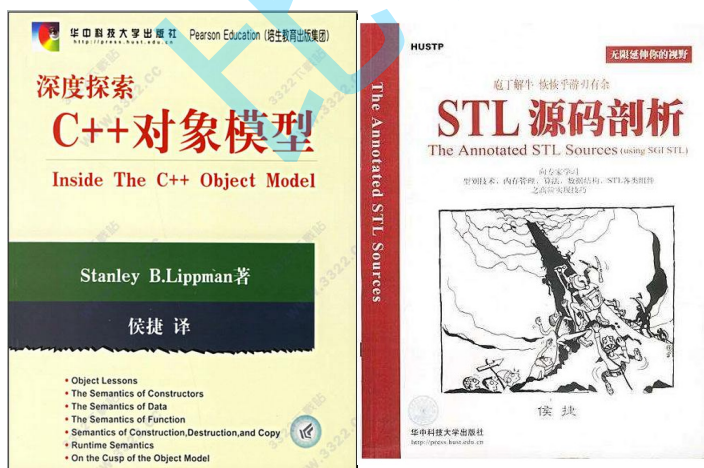
### 1. 初窥门径



### 2. 潜心修行



### 3. 深入浅出



## • 勤刷题

### 1. 牛客网

[牛客网专项训练](#)

### 2. leetcode

[leetcode中文刷题网站](#)

## 5. 关于本门课程

本门课程不仅仅是C++语法的学习，我们的目标是让大家如何高效的正常工作，因此本门课程涉及到以下内容：

- C++的基本语法
- STL库
- 高阶的数据结构

C++不仅仅是一门语言，更是一种生活哲学，在以后的工作之余还需要大家继续挖掘.....

