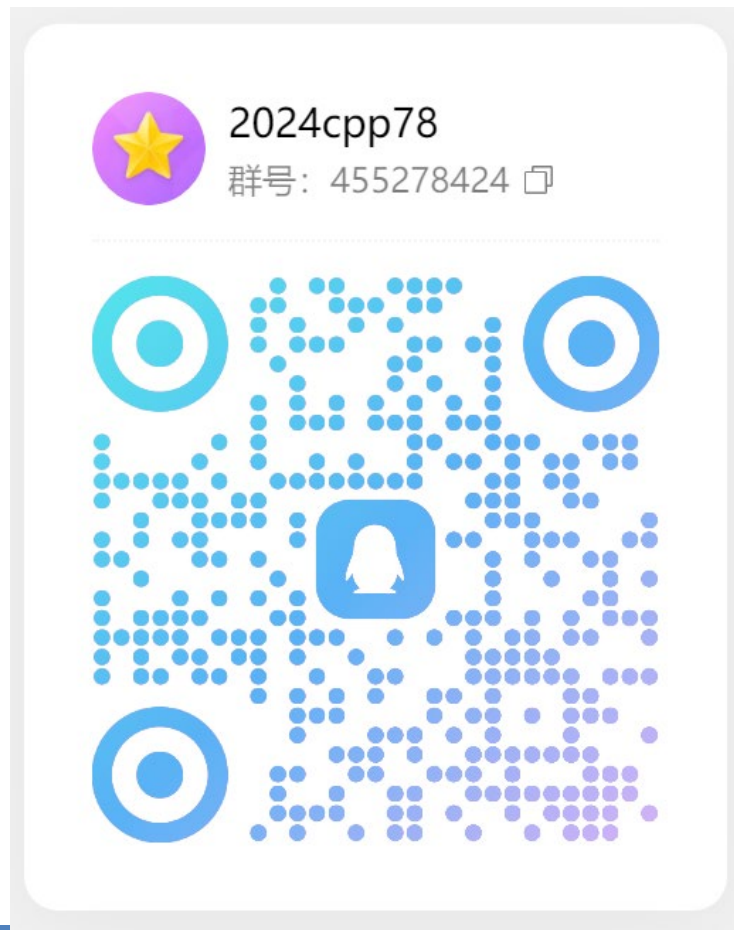


《C++程序设计》

群号：455278424



徐延宁 xyn@sdu.edu.cn
数字媒体技术教育部工程研究中心
山东大学软件学院



- 课程概述
 - 为什么学习C++
 - 本课程的内容、要求及（建议的）学习方法
- C++概述-HelloWorld

- 课程概述
 - 为什么学习C++
 - 本课程的内容、要求及（建议的）学习方法
- C++概述-HelloWorld

为什么要学习C++

为什么要学习C++

问一下
DeepSeek



已深度思考 (用时 0 秒) ^

服务器繁忙，请稍后再试。



学习C++具有多方面的价值和实际意义，尤其适合对计算机科学和软件开发有深入追求的学习者。以下是学习C++的主要理由：

DeepSeek is Great !!!

1. **高性能与效率**

- ****直接操作硬件****：C++是编译型语言，可直接管理内存和硬件资源，适合对性能要求极高的场景（如游戏引擎、高频交易、科学计算）。
- ****无额外运行时开销****：相比Java、Python等带有虚拟机的语言，C++生成的机器码运行效率更高，资源占用更少。

2. **底层控制能力**

- ****手动内存管理****：通过指针和内存分配（如`new/delete`），开发者能精确控制程序行为，优化资源利用（但需注意安全风险）。
- ****贴近硬件****：学习C++能深入理解计算机底层原理（如栈/堆、数据存储、CPU指令），为后续学习操作系统、编译原理等打下基础。

3. **广泛的应用领域**

- ****系统开发****：操作系统（Linux/Windows内核）、数据库（MySQL）、浏览器引擎（Chrome V8）等底层系统多由C++编写。



用什么开发?

C++ ? Java?

Excel? Python?

Java Script + Web UI ?

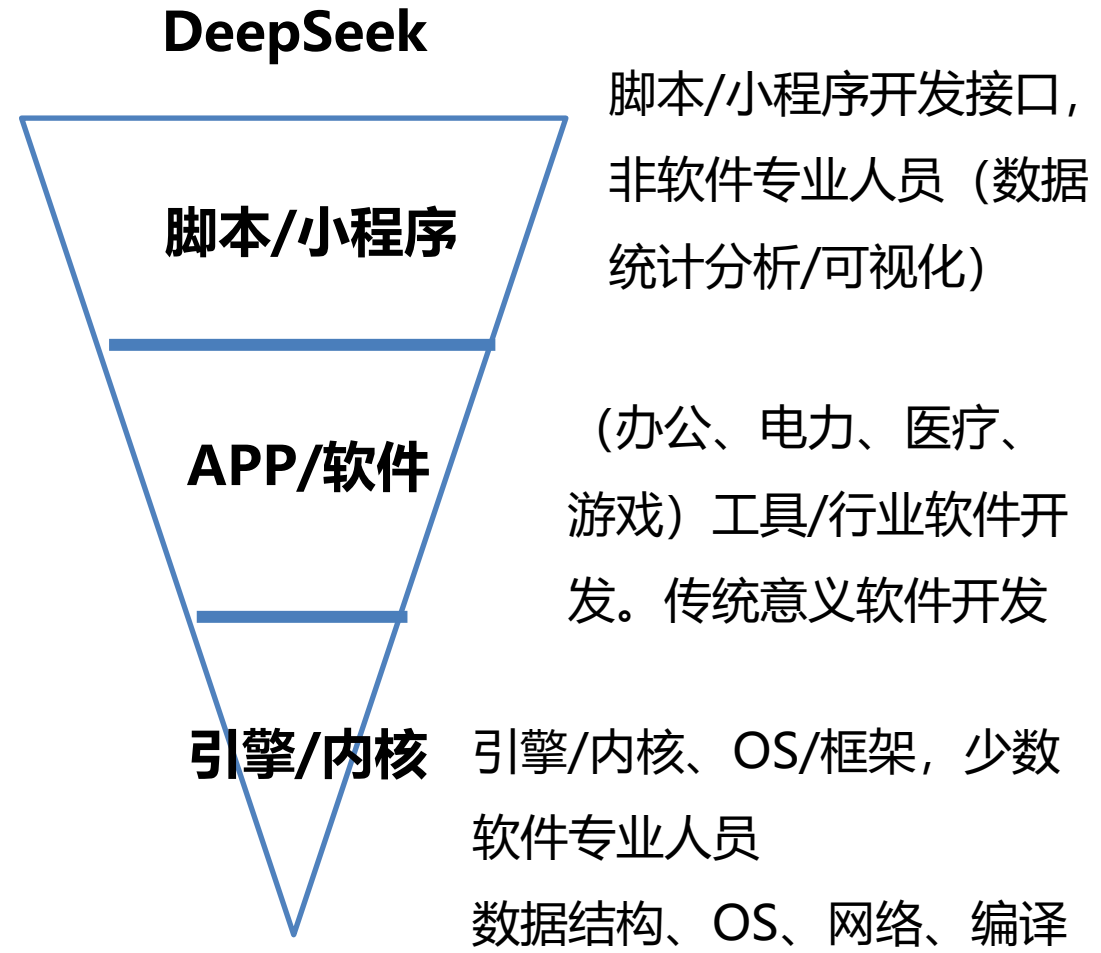
- 语言只是工具，解决问题是核心
- 软件开发者需要根据问题，选择开发工具
- 充分利用特定语言的特性，考虑**团队基础、开发效率、运行效率、可信安全、健壮稳定(模块化、可配置、可重构)**等实际问题。
 - **C++**: 从事底层算法、引擎的开发;
 - **Python+库**: 深度学习、图像视觉、数据可视化,
 - **Unity (C#) 、Unreal**: VR应用、3D游戏开发,
 - **脚本语言**: Maya、Matlab软件插件开发，微信小程序

- C++特色：编程语言之集大成者，支持开发高效、健壮、通用的程序。
 - **兼容C**：保持了C的简洁、高效，更靠近机器，能够发挥硬件的性能
 - 事实上，C++不是C，C也不是C++，在设计风格、应用场景等方面大相径庭。
 - **支持面向对象、支持泛型程序（专注于通用算法）、大量类库。**
 - Java，C#等语言在C++的基础上发展起来。
 - 代码运行效率高，自由度高：**自己管理内存，控制硬件，设计数据结构，写底层算法**
 - 运用不好适得其反，劝退了很多。
 - **主要应用于**：系统软件、编译软件、分布式系统、嵌入式系统、CAx软件、游戏（引擎）开发等等



那年我双手插兜，
不知道什么是对手

- 软件学院《**程序设计》类课程的安排：
 - 一周目：Java，必修，4.5+2学分，C++优化版：简化语法以获得更高**开发效率**；虚拟机执行，限制自由，但**安全性更高**；侧重**基于框架**的开发，（数据访问/前端UI/业务逻辑）分而治之，健壮稳定。
 - 二周目：C++，2.5学分，支撑CS (Computer Science) 数据结构、算法、编译原理、CG引擎
 - 三周目：Python与（大数据/机器学习/。。。），2学分，实际学习某某类库如何使用。Python “胶水”语言，基于丰富的模块库，一行/几行代码解决问题，**降低了开发门槛**。





- 主要内容： **C++** （语言） （与） **程序设计**
 - 标准C++语言规范：标识符，数据类型，语句，控制流程（分支、循环），函数（方法）， **数组，指针，面向对象&面向过程，模板**，基础类库
 - Thinking in C++：如何应用语言（Java C C++）特性写高效、规范的程序
 - 为数据结构、算法、编译原理等课程服务
- 面向不同领域的工程应用问题，C++包含众多第三方类库
 - 应用程序架构：MFC、QT、BOOST
 - Open GL（图形）、Open CV（图像）、Open CL（并行计算）
- **课程不包含**：C++第三方类库及其应用

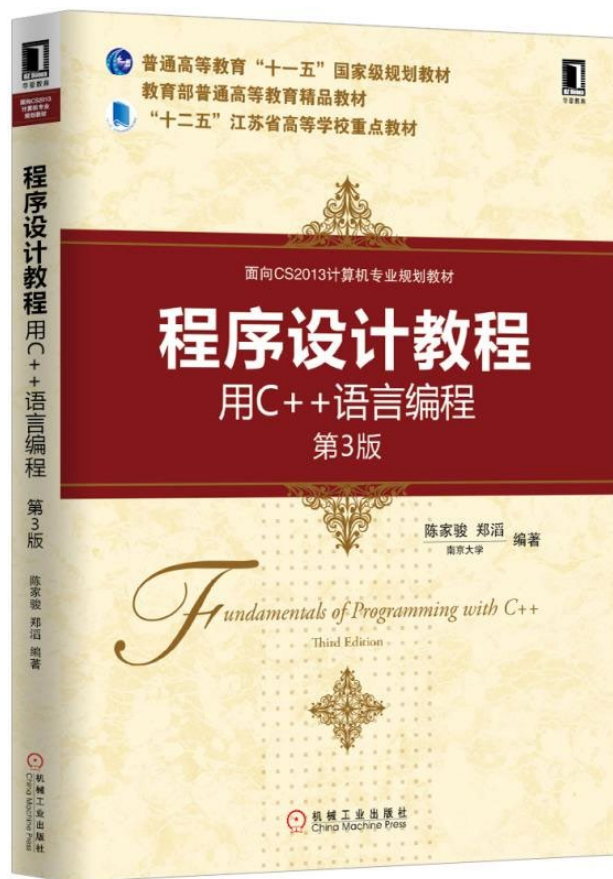


- 每周2课时 + 8次实验
- 考核：30%平时成绩 + 70%期末卷面考试成绩
- 实验：负责老师：王政
 - 四组实验作业（20分）
 - 作业质量抽查（5分）
 - 重点作业要求的实验报告（5分）
 - 实验作业：PTA平台（<https://pintia.cn>）
 - 使用过PTA的同学无需绑定
- 上机时间：第2-4、6-10周，每周五11-12节(20:00-22:00)，实验楼303机房。
 - 实验作业答疑
 - 如选修课程与实验冲突，也可参加前两班次（周五下午16:00-20:00）



- 以课件为主线进行预习复习，理解要求的知识点、例子；
 - 以二周目的视角审视C++，**课件**添加了一些Java对比、方案对比讨论。
 - **把教材作为参考资料，教材仅面向初学者的，速读教材对应章节，略过雷同情节。**
 - 超越记忆、背诵、理解的层次，自己**写代码去测试，去思考。**
- 实验作业通过PTA按时完成，学有余力多刷点题。leetCode
 - 实验，PTA自动评分对比输出判定分数（有同学直接输出，有同学不擅长输出）。
 - 有些实验用到了部分（还）没有讲的知识，怎么办，Java功底+自学能力。
- 试卷难度略低于PTA难度。
- 充分利用DeepSeek、B站等资源主动深入学习，QQ群掀起讨论的热潮（**貌似匿名实际被监管了**）

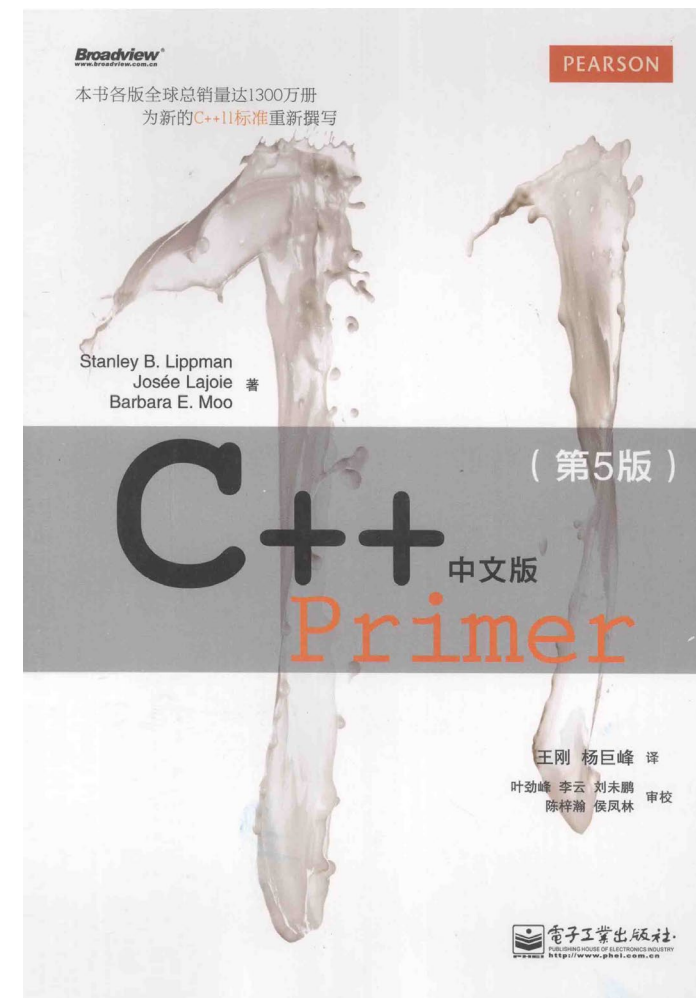
- QQ群空间：
 - 教材、参考书电子版+授课课件+原版课件：
 - 教材配套课件 & 授课课件
 - 省略一些基础，补充很多内容。每次课前都会更新，雨课堂是最终版
 - 课程教学大纲
- B站大学：视频资源（对于考试帮助不太直接）
 - C++程序设计（北京大学 郭炜、刘家瑛）
<https://www.bilibili.com/video/av10046030/>
 - C++语言程序设计（清华大学 郑莉）
<https://www.bilibili.com/video/av89313559?p=11>

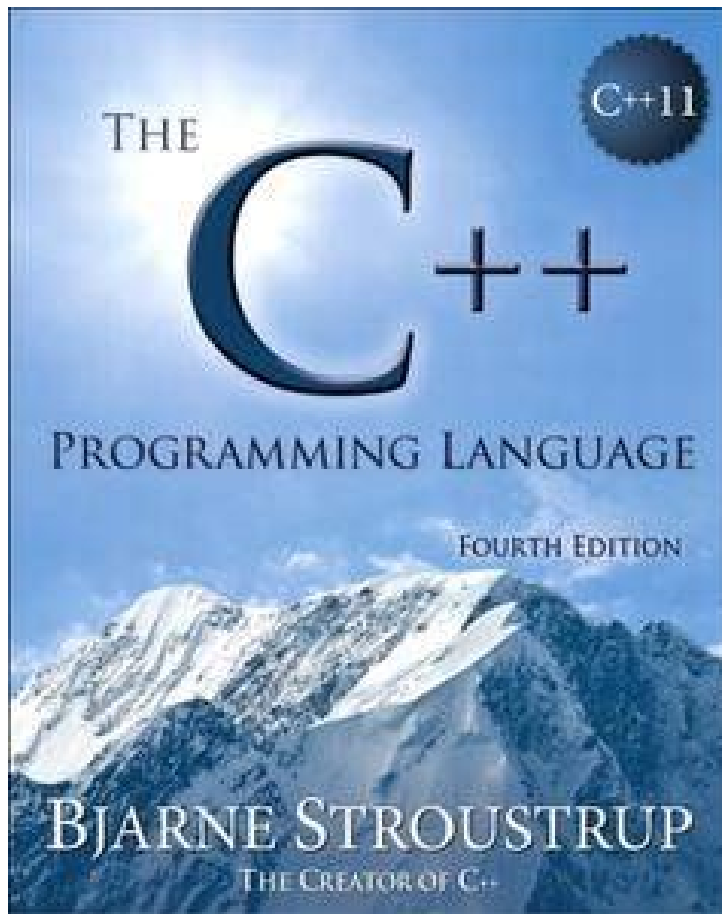


本课程主线

- 程序设计教程：用C++语言编程（第3版）陈家骏,郑滔, 2015-06-15, 机械工业出版社, ISBN:978-7-111-50123-7

C++ Primer中文版 第5版 李普曼(Stanley B. Lippman), 电子工业出版社, 2013年09月 最新版（英文版）: Edition: 6TH, 2021, Addison-Wesley Longman, Inc.





The C++ Programming Language, 4th Edition by Bjarne Stroustrup

C++之父

有电子版，中英文版
进阶书目，纯C++

本贾尼：C++之祖

我发明了 C++，制定了最初的定义，并完成了第一个实现。我选择并制定了 C++ 的设计标准，设计了大多数语言特性，设计或帮助设计了早期标准库中的很多内容，并在 C++ 标准委员会中负责处理扩展提案。

50碗鸡汤-《The C++ Programming Language》作者的经典语录： 1.把C++当成一门新的语言学习（和C没啥关系！真的。）； 3.看《The C++ Programming Language》和《Inside The C++ Object Model》，不要因为他们很难而我们自己是初学者所以就不看； 4.不要被VC、BCB、BC、MC、TC等词汇所迷惑——他们都是集成开发环境，而我们要学的是一门语言（**其实，我们学习的是描述解决问题**）；

- 课程概述
- C++概述-Helloworld
 - 第一个C++代码
 - 关于规范、编译器的那些事
 - 关于开发环境的那些事
 - 其实不重要

```
print_r('Hello World!');  
var_dump('Hello World!');  
NSLog(@"Hello World!");  
System.out.println("Hello World!");  
console.log("Hello World!");  
print("Hello World!");  
printf("Hello World!");
```

```
cout << "Hello World!" << endl;  
Console.WriteLine("Hello World!");  
fmt.Println("Hello World!");  
Response.Write("Hello World!");  
alert("Hello World!")  
echo "Hello World!"
```

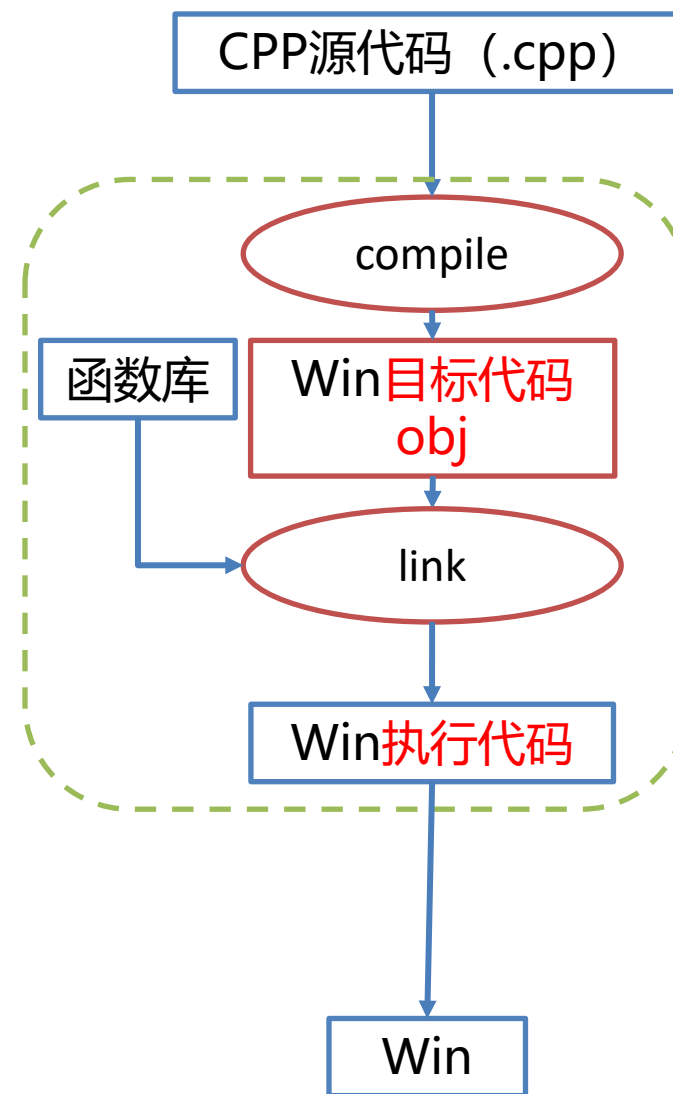
你好
萨瓦迪卡
库尼齐瓦
How are you
笨猪



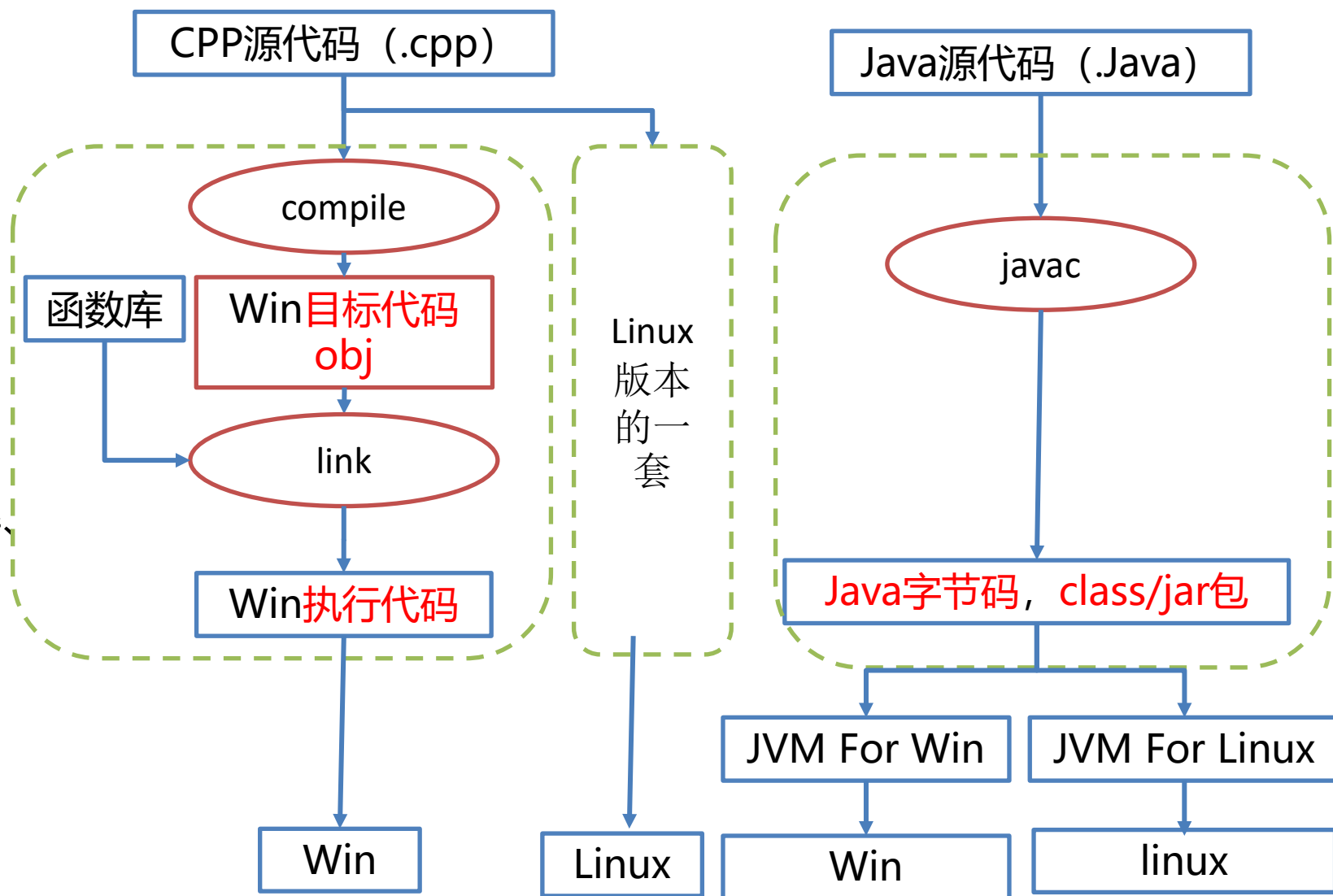
```
#include <iostream>
using namespace std;
int main()
{
    cout<<" 这是一个CPP程序."<< endl;
    system("pause");
    return 0;
}
```

- 源代码
- `***.cpp`
- `main`, 不在任何类中
- `#include` 导入
- Using `import`
- Namespace - package
- `cout`完整名字 `std::cout`
- `system("pause");` 系统暂停, 等待按键结束

- 1、编辑 Edit
 - 将源程序输入到计算机中，生成后缀为cpp的源文件。
- 2、编译 Compile
 - 以文件为单位将源代码转换为机器语言代码(目标代码)。 .obj .o
- 3、链接 Link
 - 将多个目标代码以及库中的某些文件连在一起，生成可执行文件(exe)。
- 4、运行/调试
 - Debug调试



- C++ 机器代码，直接执行
 - 优点：速度快，直接和平台交互
 - 缺点：不同平台，需要发布不同的代码
- Java 字节码（中间码），由JVM解释执行
 - 优点：不同平台有不同的JVM。JVM 负责在运行效率、安全、稳定性等方面达到一种平衡；负责跨平台。
 - 缺点：速度，灵活
- 一段C++的代码也说自己是跨平台的，怎么理解？



语言：规范/标准

C++98, C++11, C++17

Java

编译器：按照规范解释语义

GCC

Clang

MSVC

太湖之光

Oracle JDK

Open JDK

集成开发环境

TextEditor, VIM, Code Blocks, VS Code
各类编译插件

- 1973年, AT&T设计出C语言
- 1983年, Bjarne Stroustrup (本贾尼) 定义了C++, 一个更好的, **带类的C**。
- 1998年, C++被国际标准化组织 (ISO) 批准为国际标准, ANSI和ISO联合发布 "**C++98**" 标准
- 2011年, ISO标准委员会通过了新的**C++11**, **也是目前广泛使用的标准**
- C++的版本更新在不断进行中, C++14, **C++17**, C++20
- 不要再用C++11之前的版本, 尽管有些是一代经典 (计算机等级考试专用)
- C++11之后, 不必在意版本。 (不会因为使用不同语文课本而交流困难)



- 编译器是标准的支持者，不过可能不完全支持标准，可能夹带私货。
- C++编译器三足鼎立：GCC(历史悠久，推荐)、Clang（新生代，Mac, Android）、MSVC（windows，配合Visual Studio）
 - GCC太臃肿，所以有了Clang 前端编译 + LLVM（Low Level Virtual Machine）底层/后端编译
- 编译器哪家强？
 - 软硬件兼容、编译时间、执行时间
 - HelloWorld层次，不要陷入争论中
 - 我有没有机会写编译器，有：《编译原理》课程，国产机器（太湖之光），自定义语言或文件的解析
- 对于标准不明确（未定义）之处，不同C++编译器解释存在差异。
 - long 不短于 int; i+++i+++i++
 - Java 比较简明，Oracle标准的制定者与实现者，JDK一家独大，OpenJDK陪跑

- **VSCode + MinGW** (Minimal GCC for Windows) **推荐** (w64devkit)
 - 优点轻量级、万能(C C++ Java Python)、免费的编辑器。 一个目录下好多带main的cpp, 每个都可以生成exe, 想运行哪个运行哪个。其他IDE一个工程只能有一个文件包含main
 - 缺点: **DIY配置**。配置安装方法问百度: 关键词 VS Code + MinGW配置 + **视频**
 - <https://code.visualstudio.com/docs/cpp/config-mingw>, 官方教程
 - **配置**可能需要一上午的时间, 总有一些意外, 例如包含了中文路径名。
 - 将来: 编译一个开源系统, 可能折腾*周, 东市买骏马, 西市买鞍鞯。
- Code::Blocks: <https://sourceforge.net/projects/codeblocks/>
- Devcpp: <https://sourceforge.net/projects/orwelldevcpp/>
- Visual Studio Community, **无需配置**
- 没有要求, 随意选用。。。CLion, XCode, VIM, 记事本 (PTA)



- 安装编辑器VSCode
- 配置开发包 (GCC)
- VS**插件**连接二者 (生成配置文件), 创建工程
- C++程序验证
- 附加说明
- PATH环境变量的作用
- 技术选型的依据

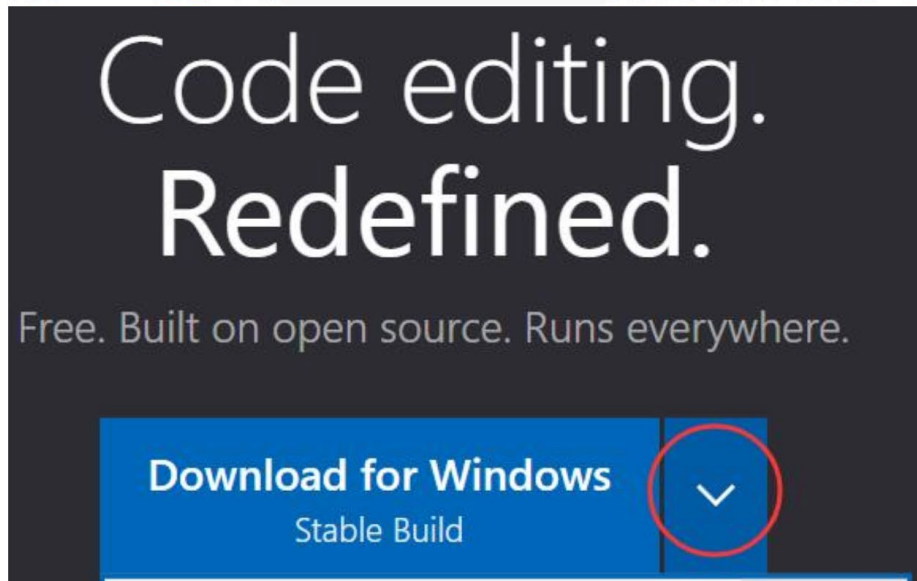
C++开发环境配置 (Win版)

这里为使用Windows的同学介绍一种C++开发环境的搭建方法。开发环境总要有编辑器 (editor) 用来写源代码, 需要编译器 (compiler) 把源码翻译为可执行程序。编辑器选用Visual Studio Code (简称VSCode), 是非常流行的开发工具。编译器选用 w64devkit 开发包, 内含GCC-13.2, 也基本是最新版本, 支持C++20和部分C++23语法特性。最后有这套方案选型依据的说明。

第1步: 安装编辑器

VSCode是微软出品的轻量编辑器, 扩展性出众, 可配置为多种语言的IDE。

1. 浏览VSCode官网, 下载 Windows x64 User Installer 安装包。如果下载很慢, 可从学校云盘下载, 参考课程QQ群公告。

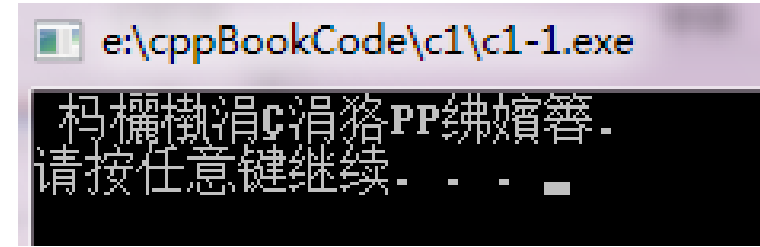
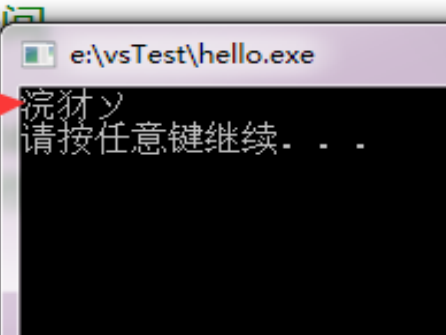


<https://gitee.com/sduwz/cpplab/>

```

1  #include <iostream> // 输入输出流类库
2  using namespace std; // 名称空间
3  int main(){
4      cout <<"你好"<<endl; // 输出
5      system("pause"); // 结束关闭
6      return 0;
7  }

```



```

"tasks": [
    {
        "command": "C:\\mingw-w64\\mingw64\\bin\\g++.exe", // 本质用g++编译
        "args": [
            // g++的命令行参数
            "-g", // 产生调试信息
            "${file}", // 源文件

```

// "-finput-charset=UTF-8", // vs code源代码默认为UTF-8，即编译器g++接受的输入为utf-8，如果是自己写的代码，请使该缺省配置

// "-finput-charset=GB2312", // 课本提供的源代码编码为GB2312，如果运行课本源代码，需要告知mingw，源代码是gb2312编码的

```

//
        "-fexec-charset=GBK", // 友情赠送，Windows中的cmd console中文编码为GBK
        "-o", // 后面是，目标文件
        "${fileDirname}\\${fileBasenameNoExtension}.exe" // 目标文件
    ], // 以下忽略
}

```




```
#include <iostream>
using namespace std;
int main(){
    cout << "sdu你好xyn" << endl;
    system("pause");
    return 0;
}
```

Launch.json "externalConsole": true,

sdu你好xyn
请按任意键继续. . .

Launch.json "externalConsole": false,

sduxyn
Press any key to continue . . .

- 汉字编码不在C++语言标准内，所以各个编译器处理方式不同。
- minGW的GDB console与Window console的汉字编码不同。 . . .

- C++新手：我就是写个helloworld，为什么要知道这些？痛苦
- Java程序员：字符串Unicode编码，有效解决了汉字显示，Java真香。
- **C++老手：汉字输出不重要，有趣的灵魂胜过好看的皮囊，Java的GUI只是样子货。**



- 配置好系统，写个小练习=》
- 如果HelloWorld还有问题
 1. 编译器是否安装好了？c++ -version
 2. 如果编译没通过，没生成exe，则是tasks.json的问题，通过命令行验证下 g++，参数是否有问题
 3. 如果生成了exe，没法调试，则是launch.json问题，gdb配置问题，确认下path以及miMode
 4. 问百度、问B站、问王老师，问同学，问自己长得帅？

输入十个整数，打印输出其中的最大数值。

```
#include <iostream>
using namespace std;
int main() {
    int a, m;
    cin >> a;
    m = a ;
    for (int j = 1; j < 10; j++) {
        cin >> a;
        if (a > max) max = a;
    }
    cout << "max=" << m << endl;
    return 0;
}
```

- **Next: 简单C++程序: 数据类型与语句**
- 函数
- 数组与指针
- 自定义数据类型
- 类和对象使用
- 运算符重载
- 继承和派生
- 模板与容器、输入输出、异常

C++开发环境，我选择了

- ☐ A Visual Studio C++
- ☐ B VSCode + w64devkit
- ☐ C Code::Blocks 或者 DevC++
- ☐ D JetBrains的CLion
- ☐ E Xcode或者其他开发环境
- ☐ F 我还没有开始或者没有成功编写HelloWorld

提交

谁被誉为C++之父



A

Bjarne Stroustrup

C

Guido

B

James Gosling

D

Muyu

提交