C++

Programming

C++初步知识

Preliminaries of C++

2025年2月17日

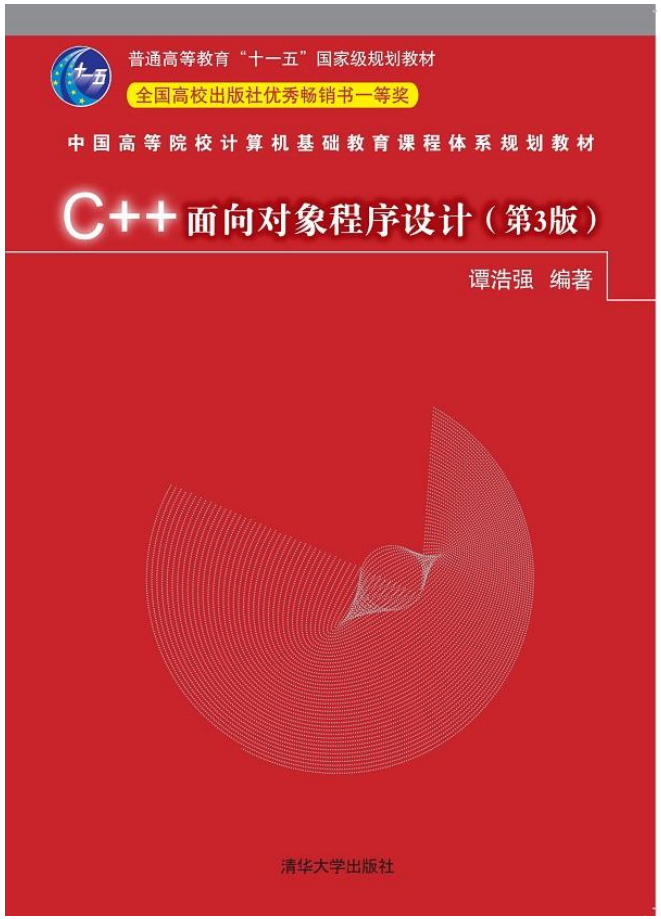
学而不厌 誨人不倦

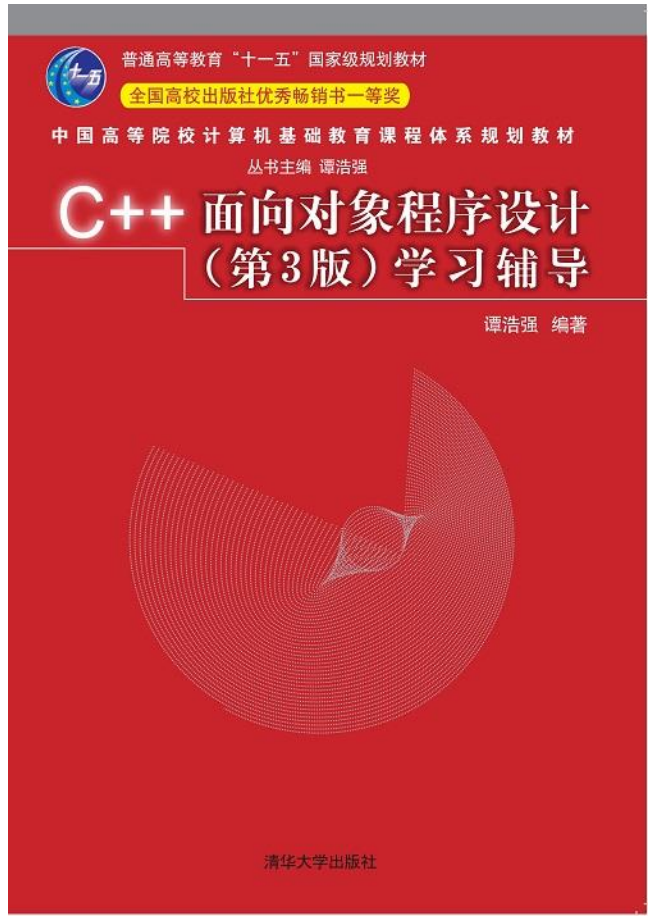
课程简介

➢ 教学内容

课程代码及名称
301039 C++程序设计

第1章 C++的初步知识
第2章 类和对象的特性
第3章 怎样使用类和对象
第4章 对运算符重载
第5章 继承与派生
第6章 多态性与虚函数
第7章 输入输出流
第8章 C++工具





主要参考教材

教学内容与考核

➢ 课程目标

1. 了解课程和专业相关前沿
2. 熟练掌握C++编程基本技能：编码、调试

- Learn how to write **powerful** and **elegant** code.
- Write **actual** C++, no libraries
- Understand the design decisions that lead to "**good**" code

3. 掌握利用C++解决实际问题的初步能力

➢ 考核方式

1. 课堂教学30%：出勤、准时、实验作业、主动提出问题等
2. 课程大作业 40%： 预计第4周布置题目，第10周前完成
3. Spotlights 20%：第10-12周完成 每位同学PPT讲解

➢ 先修/后续课程

C 程序设计→C++面向对象程序设计→数据结构/算法设计

年级-专业-姓名

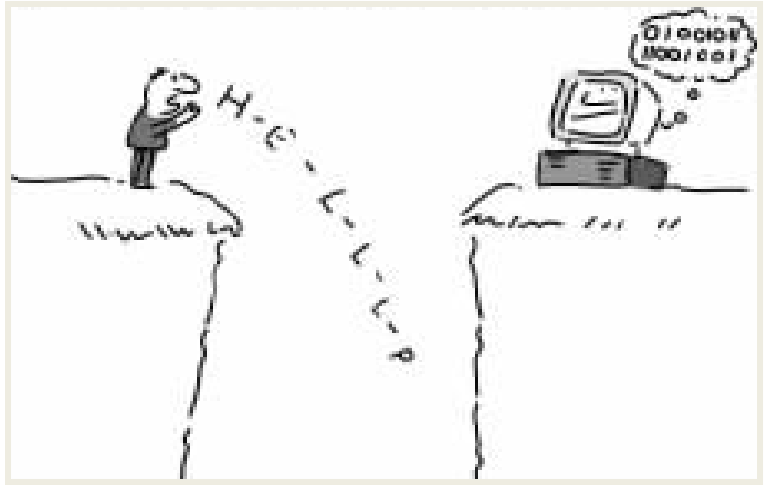
程序的作用

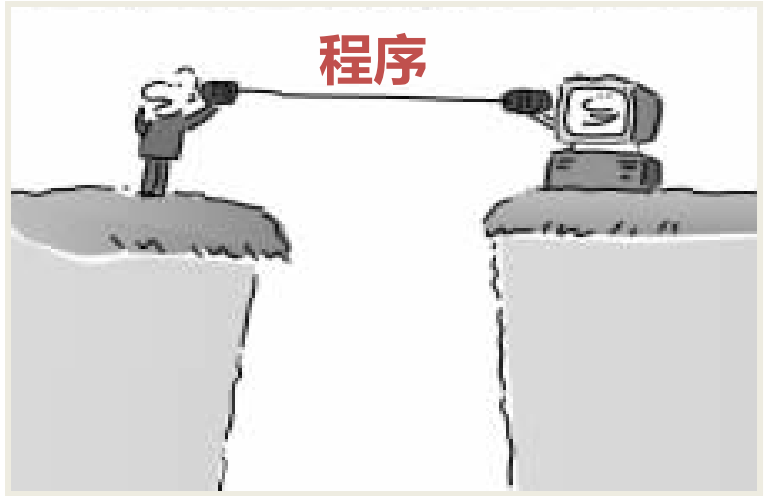
为什么要写程序？程序有什么用呢？

冯.诺依曼结构：

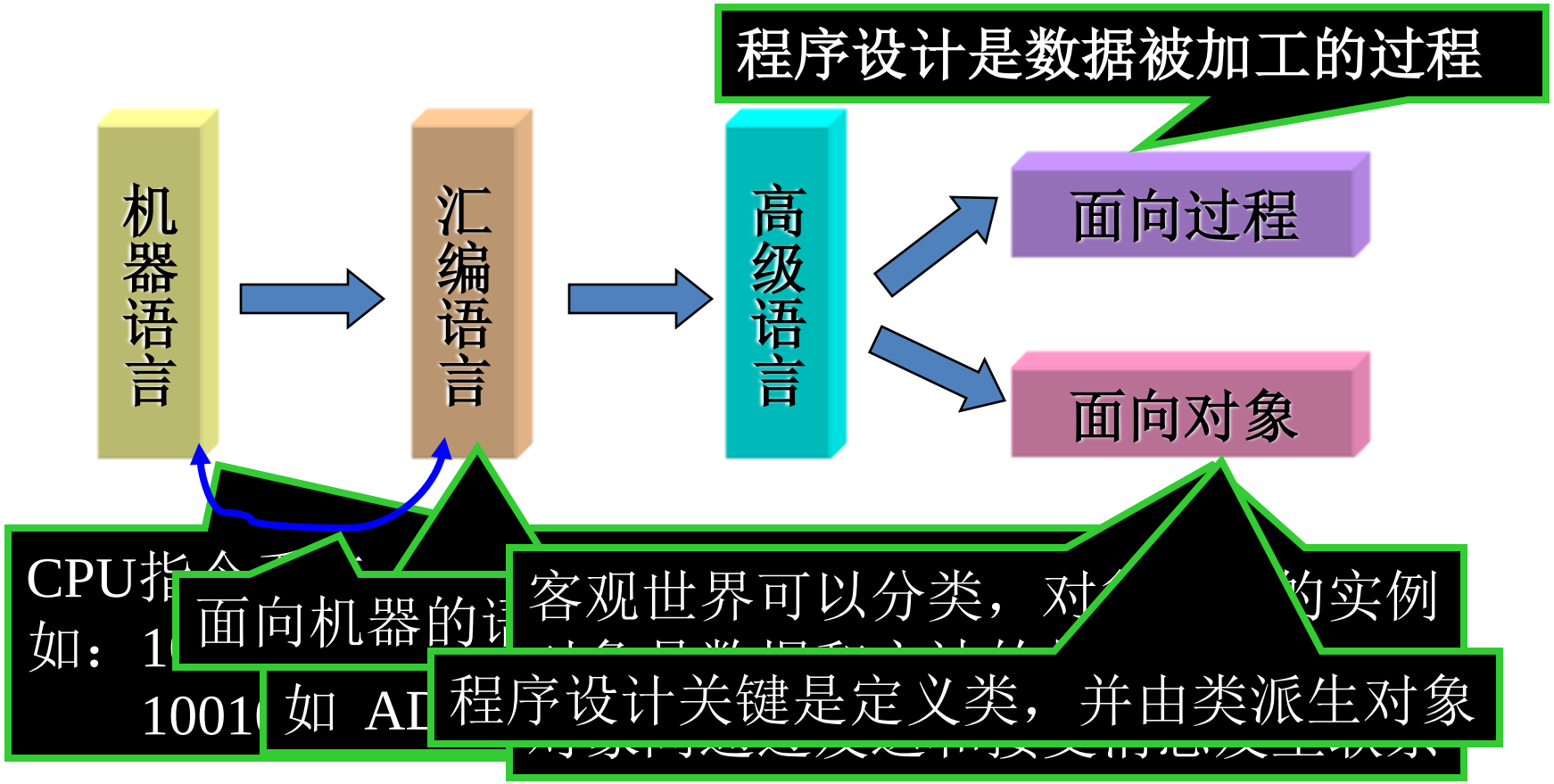
- 计算机
  - 主机：
    - 运算器
    - 控制器
    - （运算器、控制器 → 中央处理器CPU）
    - 存储器
  - I/O设备：键盘、显示器等

人要和计算机有效地交流，必须通过程序

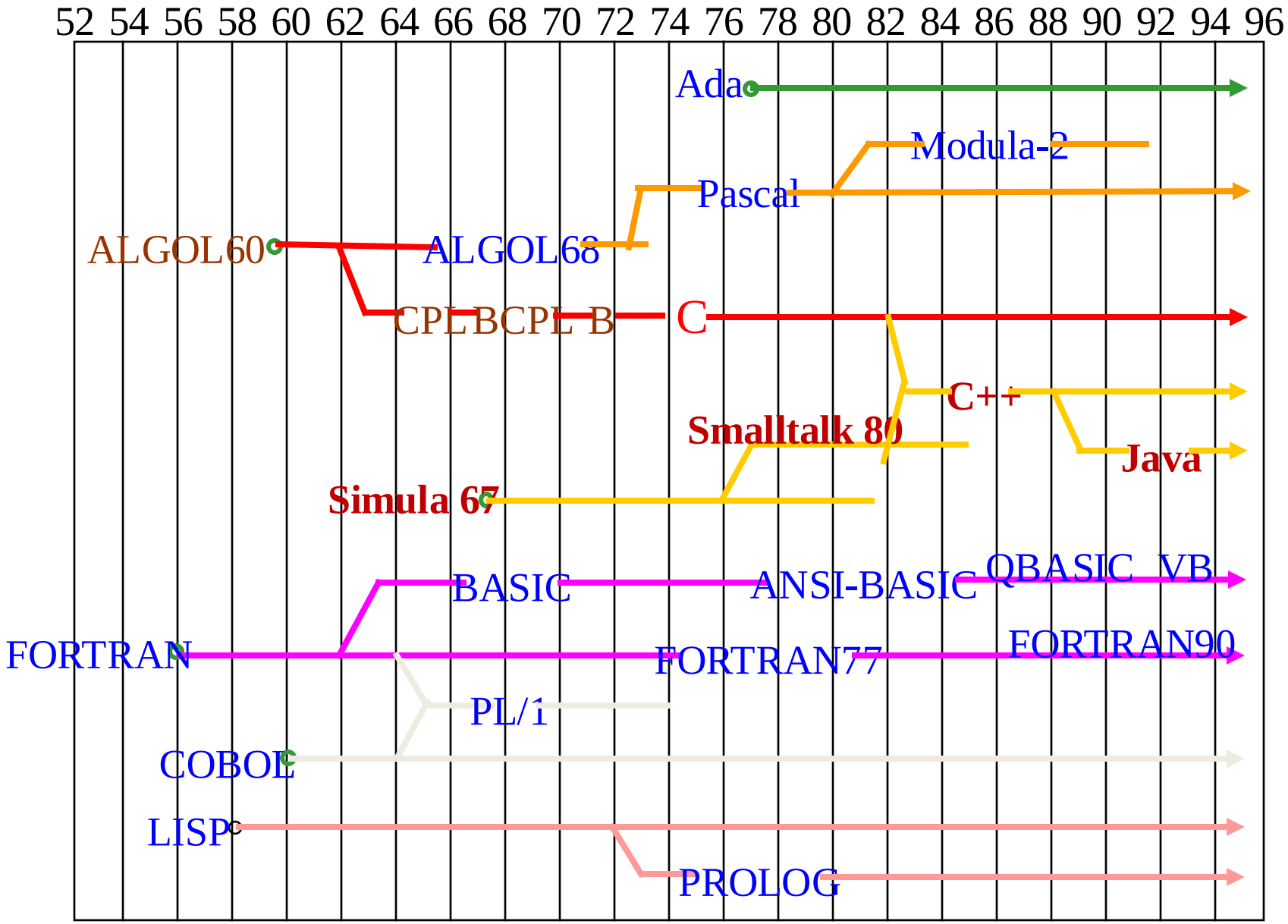




程序设计语言


程序设计是数据被加工的过程
机器语言
汇编语言
高级语言
面向过程
面向对象
CPU指令
如：1
1001
如 AD
面向机器的
客观世界可以分类，对
的实例
程序设计关键是定义类，并由类派生对象

程序设计语言


52 54 56 58 60 62 64 66 68 70 72 74 76 78 80 82 84 86 88 90 92 94 96
Ada
Modula-2
Pascal
ALGOL60
ALGOL68
CPL
BCPL
B
C
C++
Smalltalk 80
Java
Simula 67
BASIC
ANSI-BASIC
QBASIC
VB
FORTRAN
FORTRAN77
FORTRAN90
PL/1
COBOL
LISP
PROLOG

Very brief history of C++

1962
Simula I was invented by

1969-1973
The C language was invented

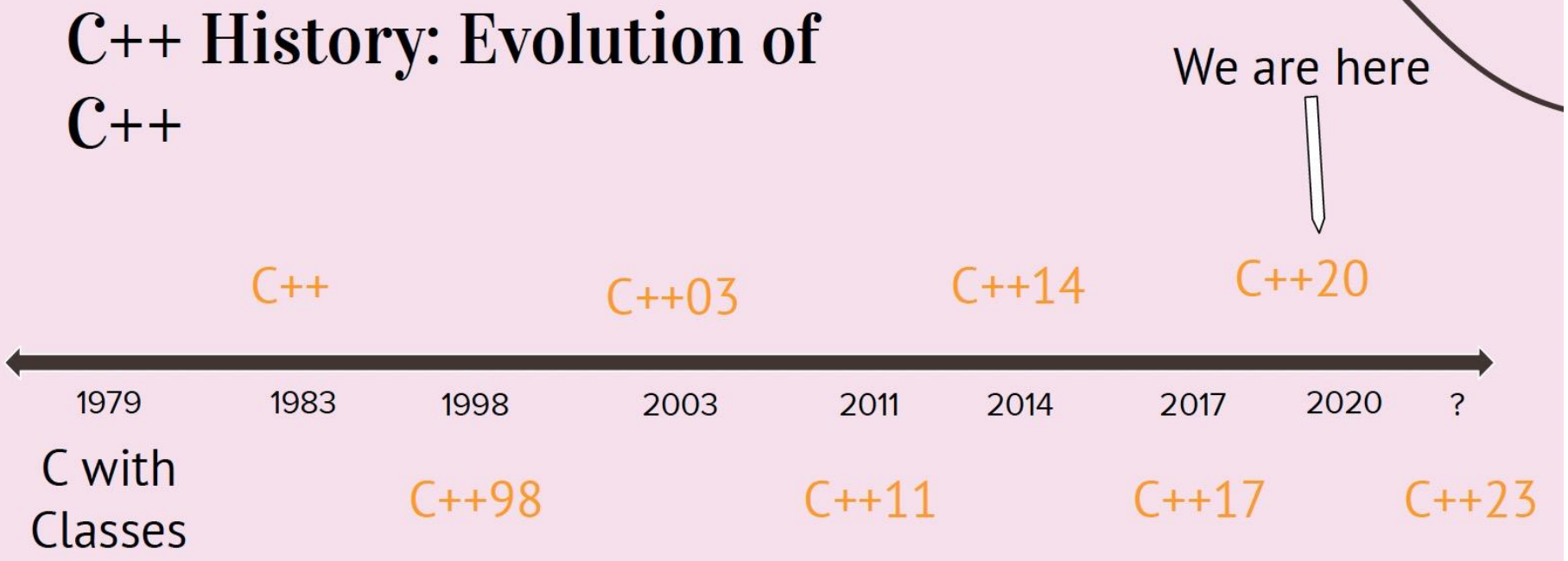
2014
Minor update: C++14 released.

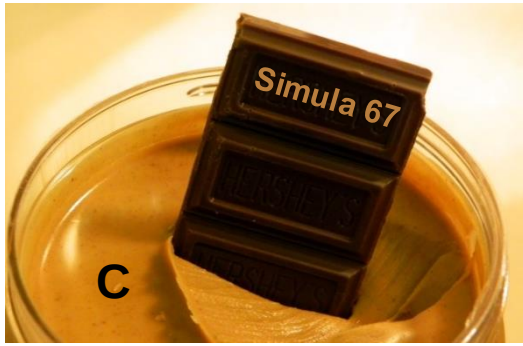
Bjarne Stroustrop began developing "C with Classes"

"C with Classes" renamed to C++

1st commercial C++ compiler, Cfront, released by AT&T

C++ History: Evolution of C++





C++

For details more check out [A History of C++: 1979−1991]

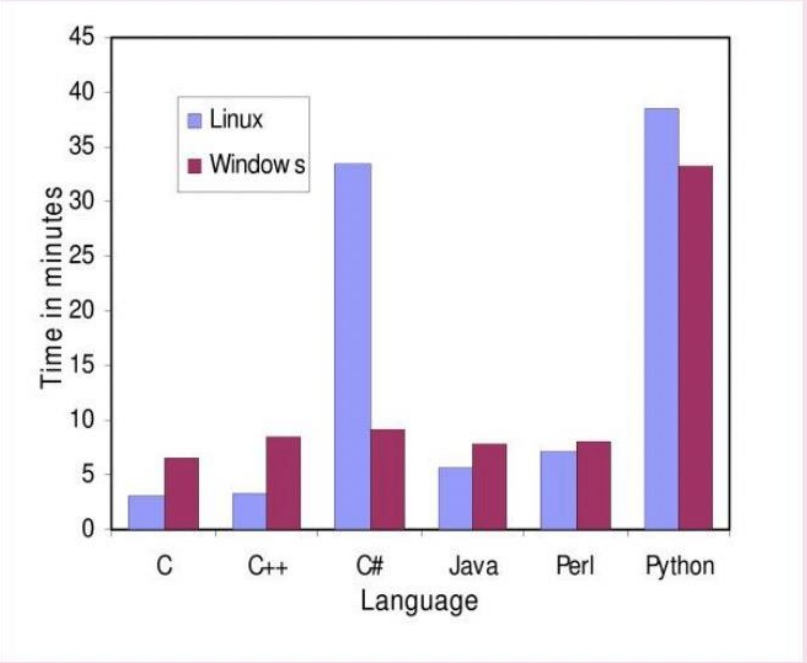
Why C++

Many Cool Things Use/Were Made with C++

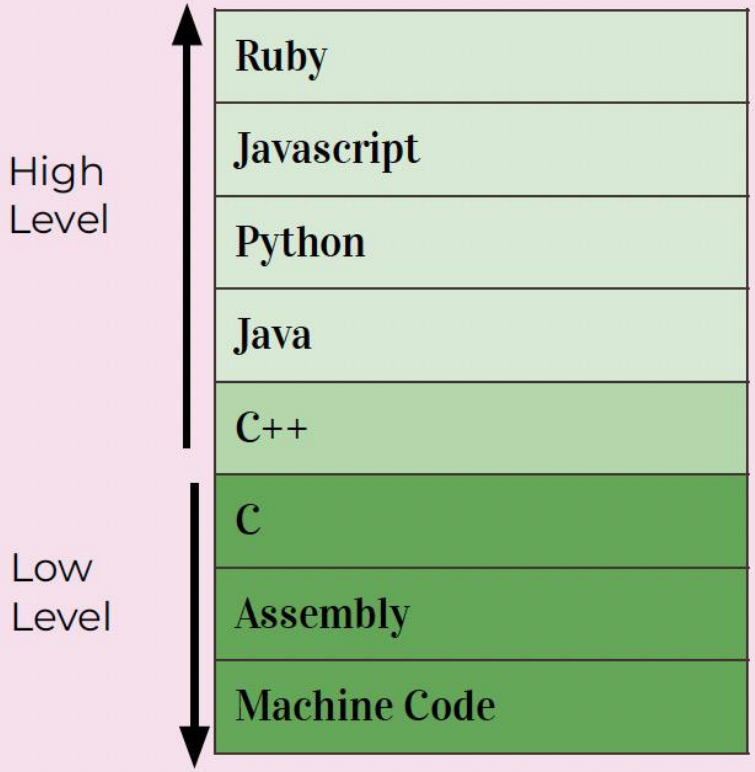
amazon

Why C++?

FAST

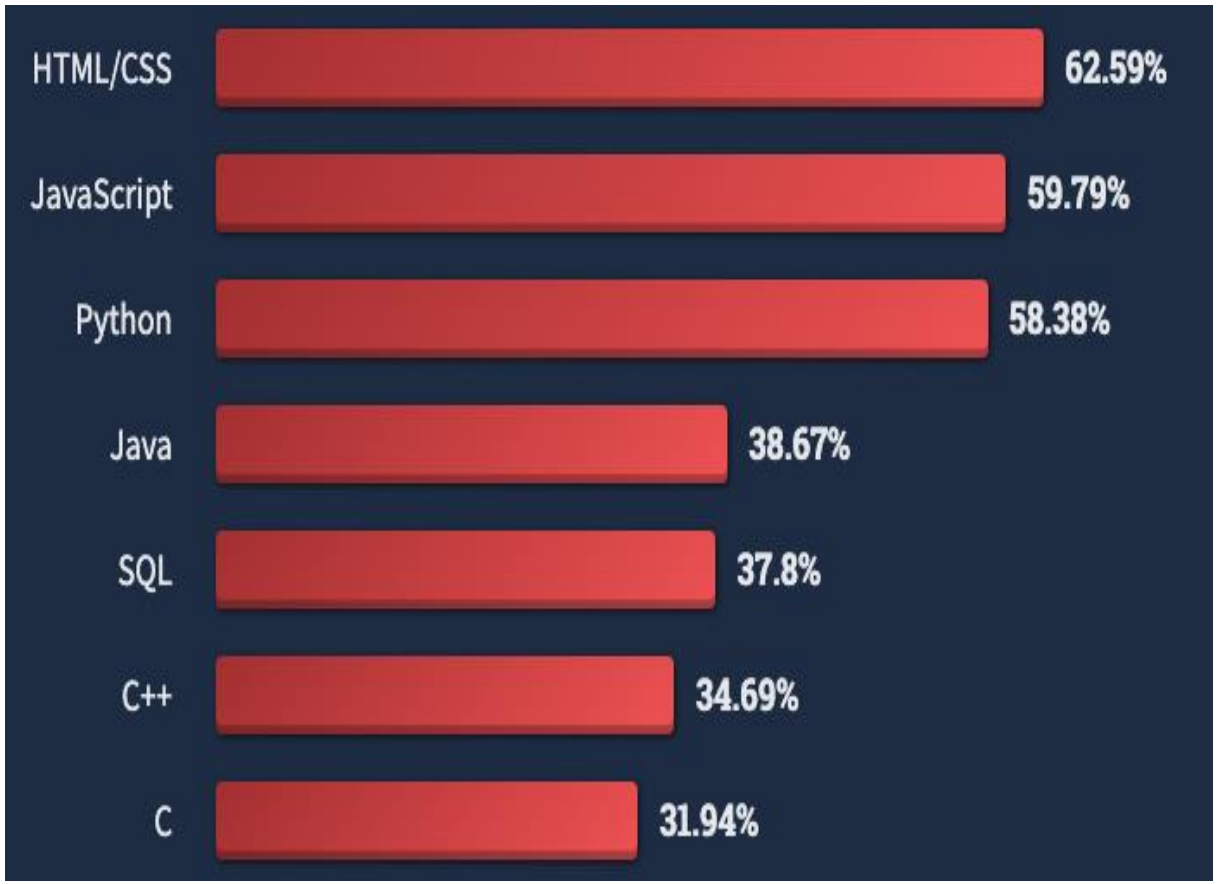


Lower-level control

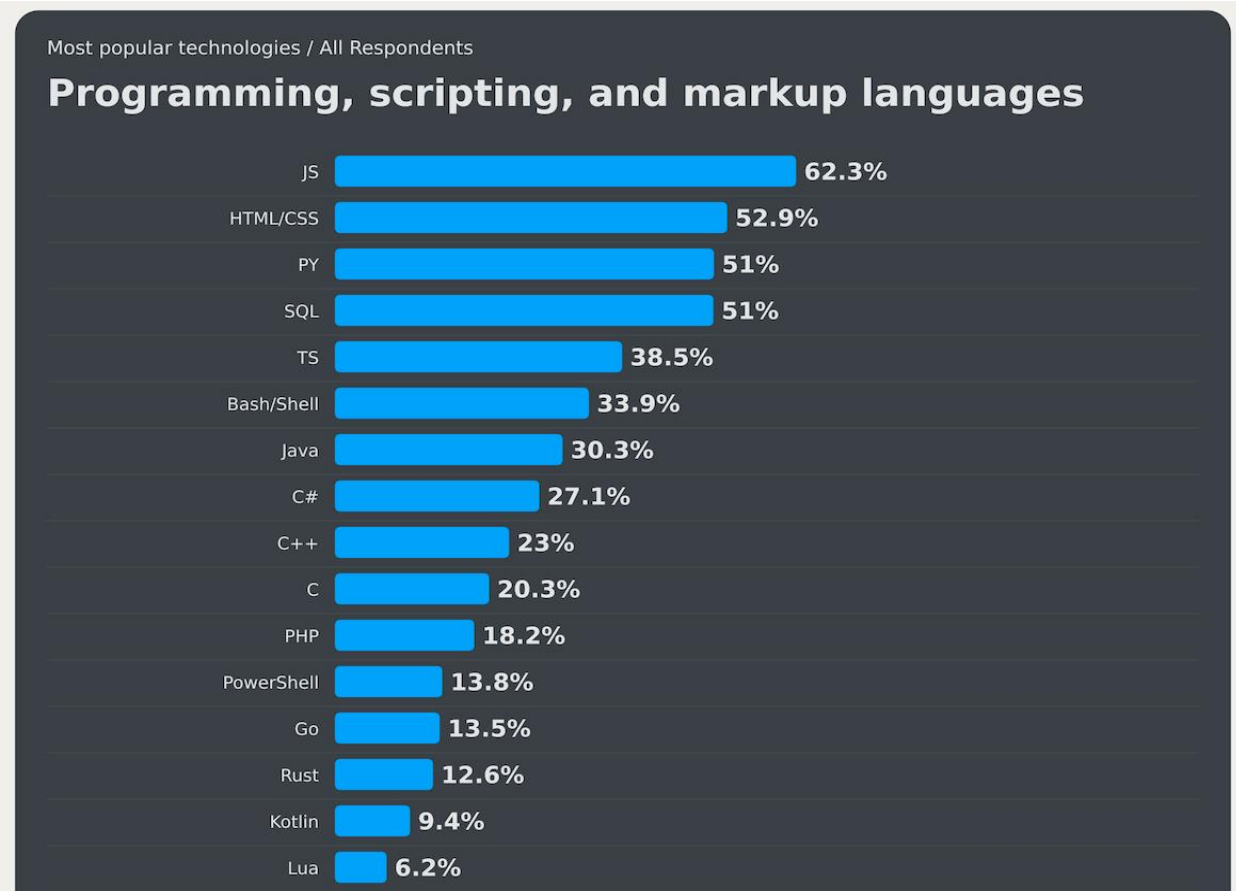


Why C++

The most popular programming languages for novice programmers



https://survey.stackoverflow.co/2022/#most-popular-technologies-language-learn



https://survey.stackoverflow.co/2024/technology#1-databases

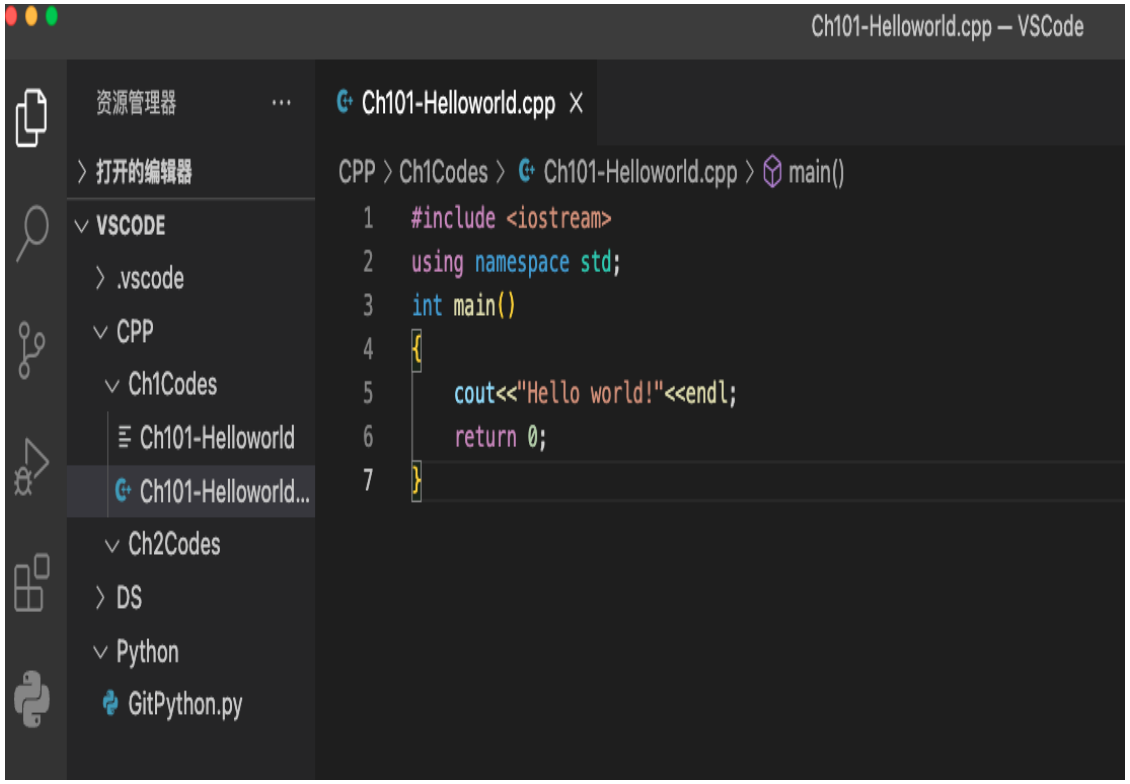
https://www.adesso.de/en/news/blog/stack-overflow-developer-survey-2022-part-2-2.jsp

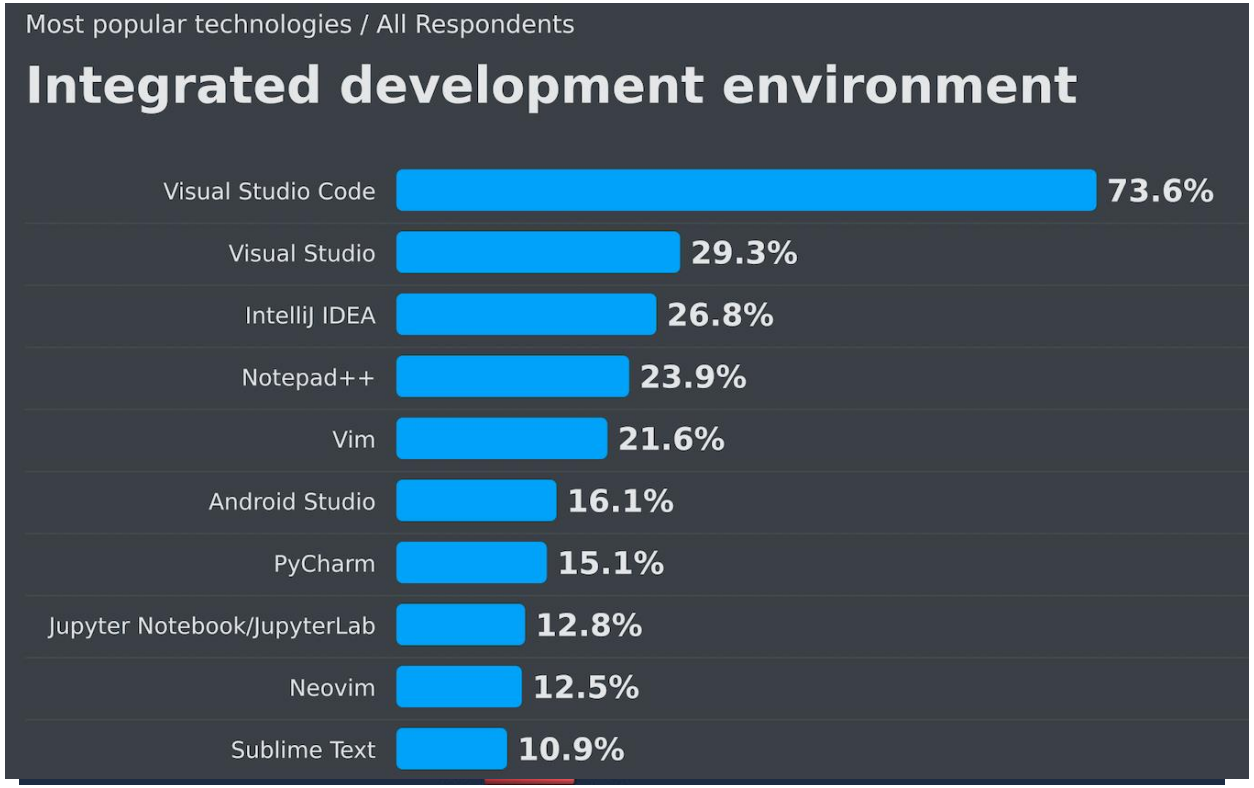
Why C++

https://www.unitree.com/cn/position

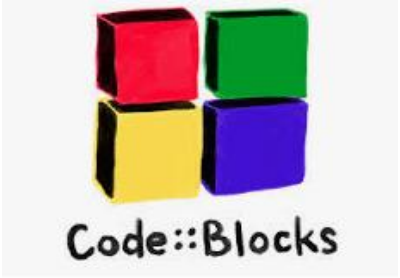
关于编程工具

VSCode / Code::Blocks / Microsoft Visual Studio 6.0/LatestVersions





https://survey.stackoverflow.co/2022/#most-popular-technologies-new-collab-tools



关于开放课程资源

CS 106L

Standard C++ Programming

Stanford University, Winter 2023

About CS106L

CS 106L is a companion class to CS106B/CS106X that explores the modern C++ language in depth. We'll cover some of the most exciting features of C++, including modern patterns that give it beauty and power.

Anyone who is taking or has taken CS 106B/X (or equivalent) is welcome to enroll. In other words, we welcome anyone that has learned or is learning programming fundamentals like functions and objects/classes.

CS 106L is a class for **1 unit**. Students will complete 2 ***very short*** assignments. There are ***no exams or papers***. All grades are S/NC. Class will finish in week 9 to give you time for finals.

CS 106L is built for you! Even if you're not taking the class, you're welcome to come to our in-person office hours. Haven: Tuesdays 4:30 - 5:30pm in 260-113. Sarah: Wednesdays 3:15 - 4:15pm in 120-314. If these times don't work for you, we'll also have **virtual** office hours, times by appointment!

Course Information

Haven Whitney

Sarah McCarthy

cs106l-win2223-staff@lists.stanford.edu

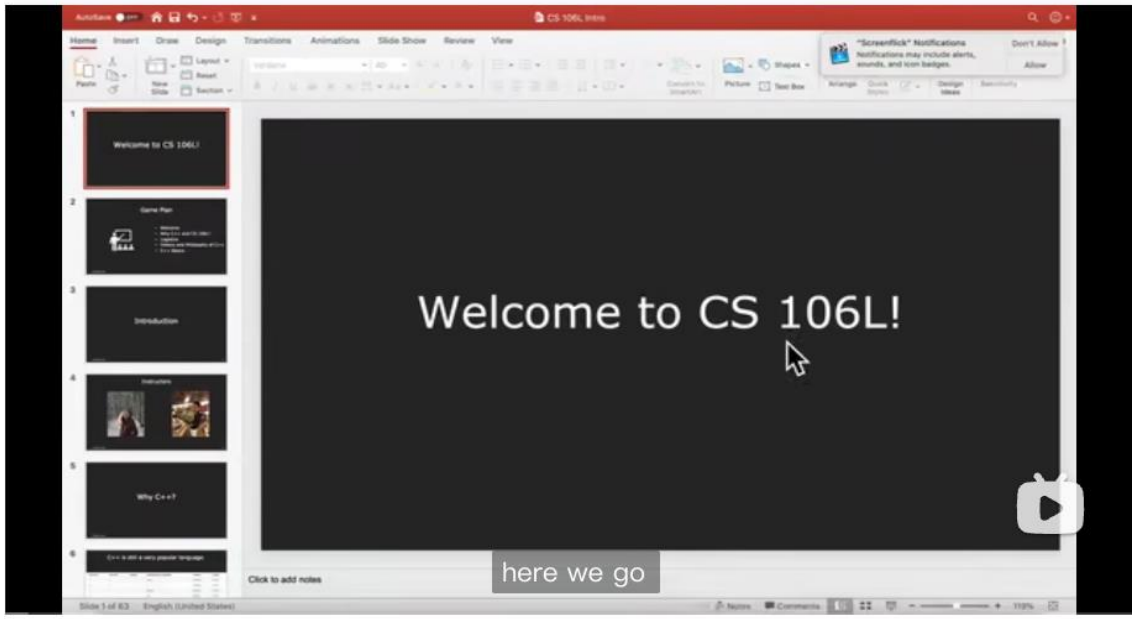
Tuesday, Thursday; 3:00-4:20pm; 260-113

Week	Tuesday	Thursday
1	JANUARY 10 1. Welcome! Slides Policies	JANUARY 12 2. Types and Structs Slides Code
2	JANUARY 17 3. Initialization and References Slides Code	JANUARY 19 4. Streams Slides Code
3	JANUARY 24 5. Containers Slides	JANUARY 26 6. Iterators and Pointers Slides Code

http://web.stanford.edu/class/cs106l/

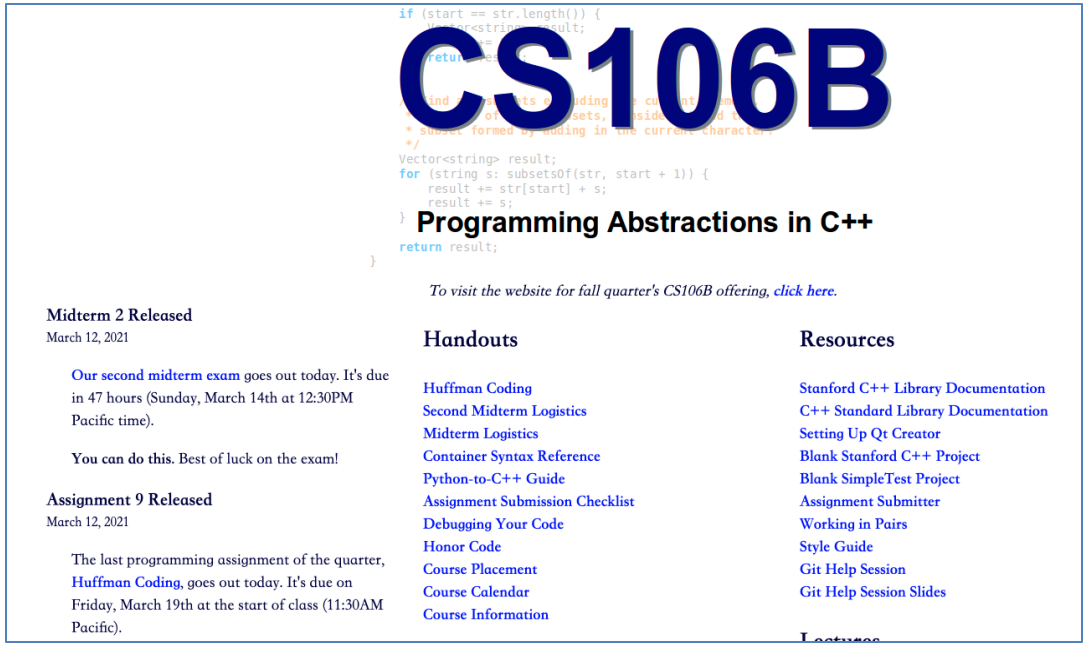
【中英双语🎉】CS106L: Standard C++ Programming, Special Edition

1.3万 11 2022-09-17 20:19:58



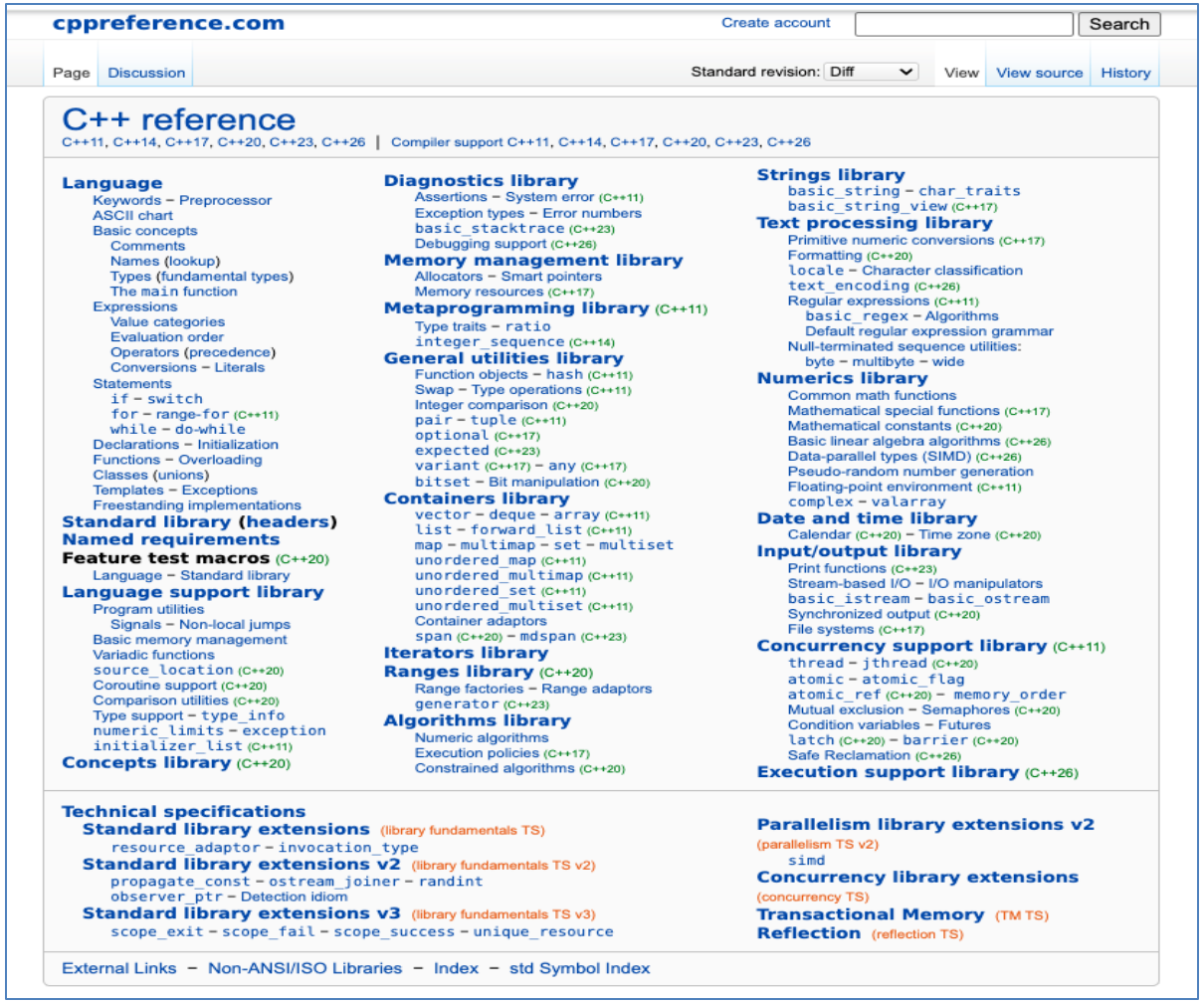
https://www.bilibili.com/video/BV1K8411b7AU/?spm_id_from=333.337.search-card.all.click

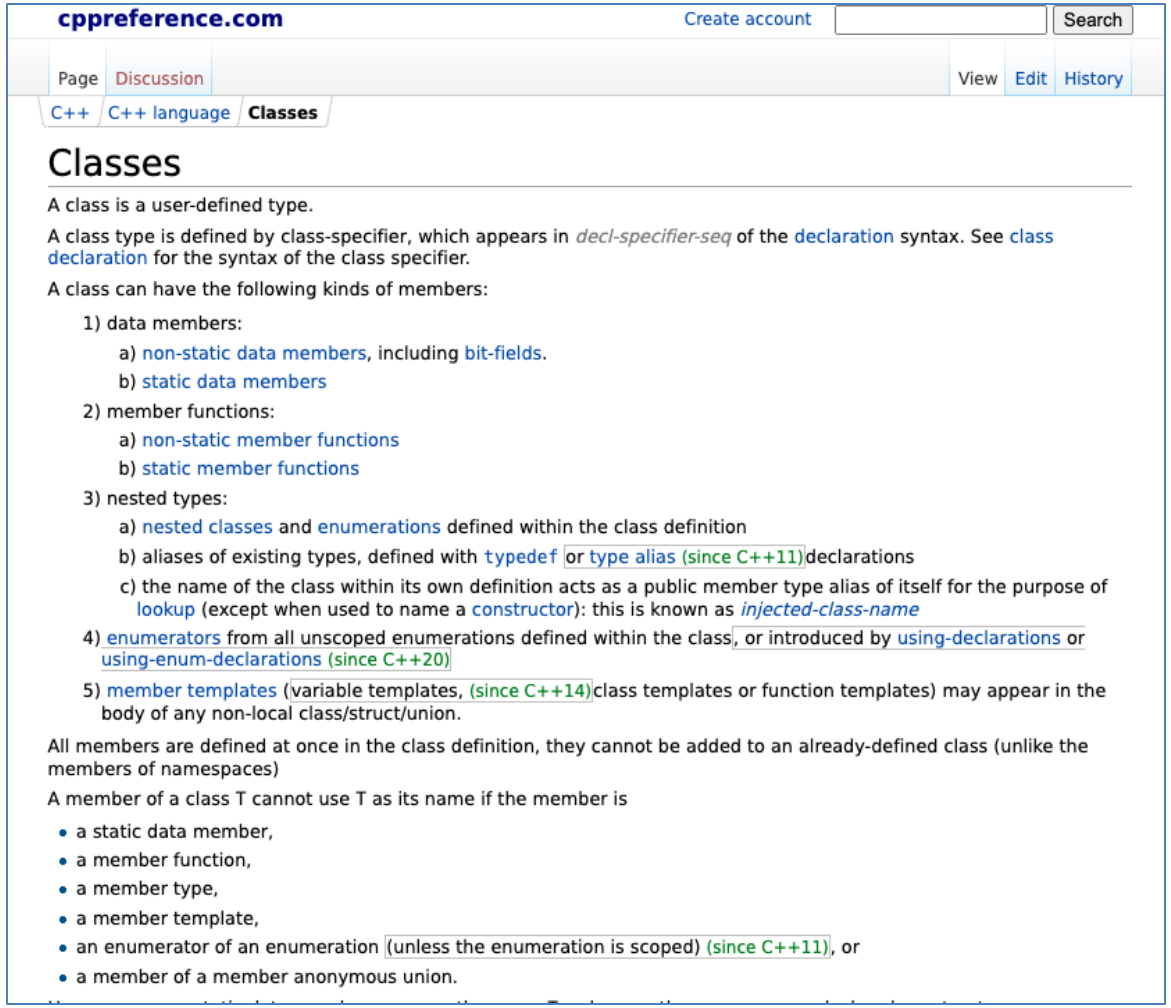
关于开放课程资源



- **课程教材：** https://web.stanford.edu/class/cs106x/res/reader/CS106BX-Reader.pdf
- **课程视频：** https://www.bilibili.com/video/BV1G7411k7jG

关于课程资源





https://en.cppreference.com/w/

1.1 从 C 到 C++

POP： Procedure-Oriented Programming

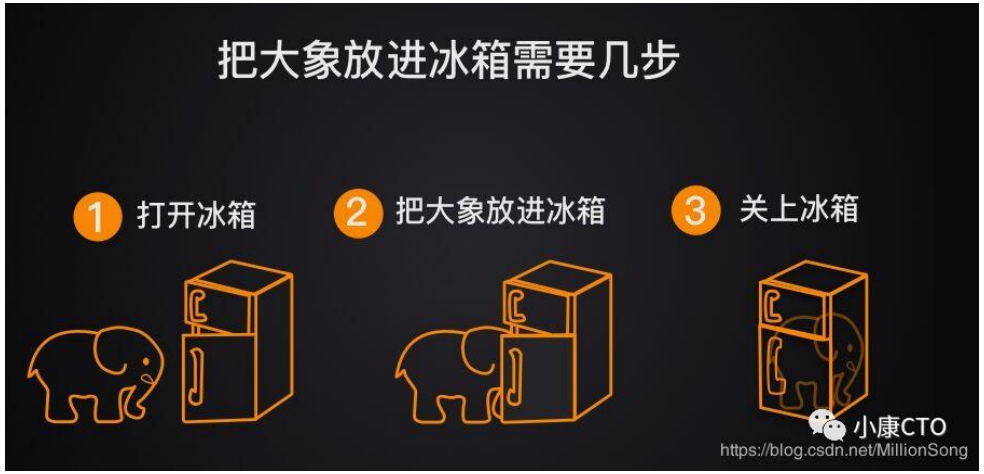
➢ 面向过程编程

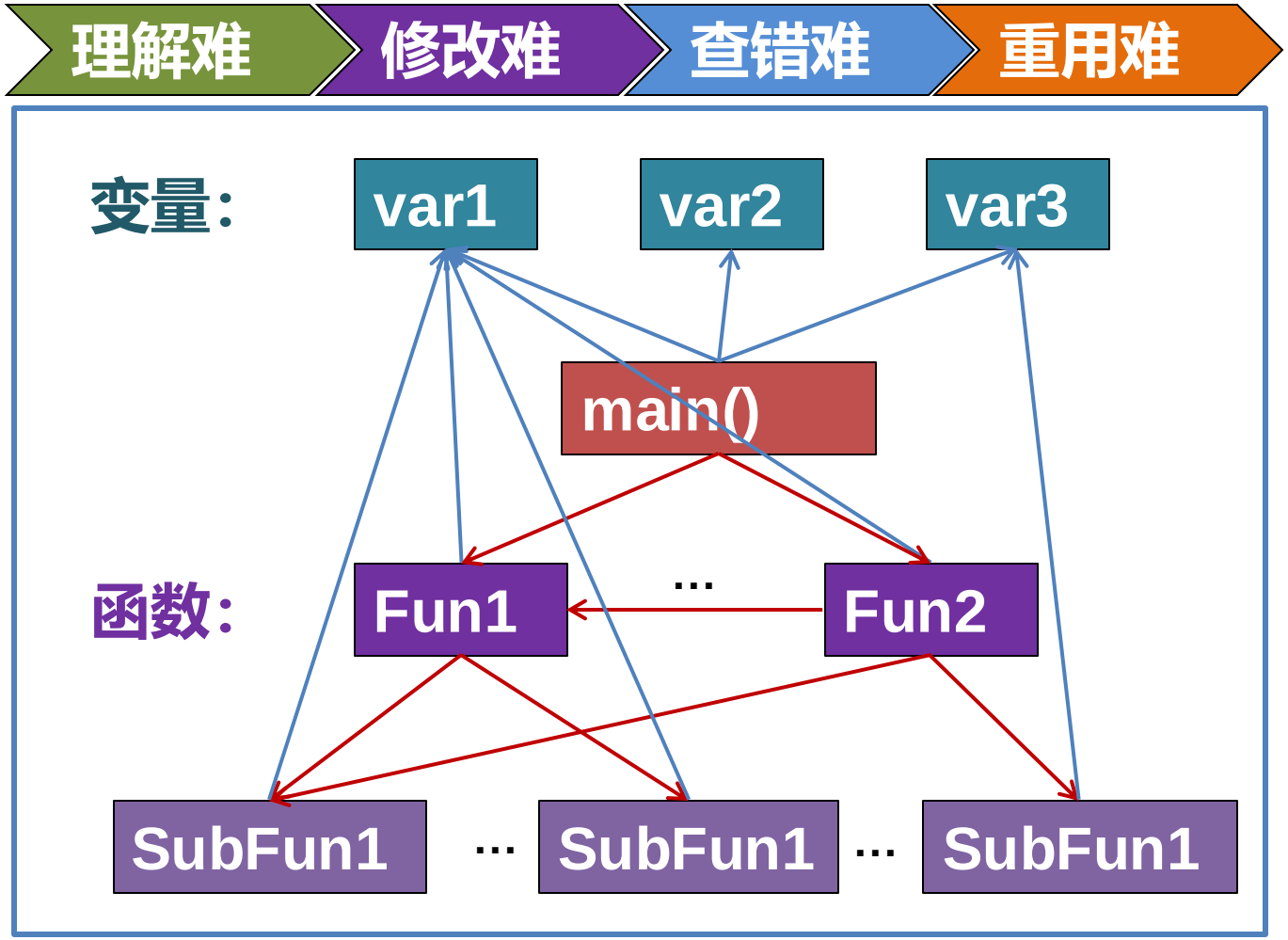
按部就班、自顶向下的编程

模块化编程：**函数+变量=程序**

常量、变量、指针、结构体

顺序、选择/分支、循环结构

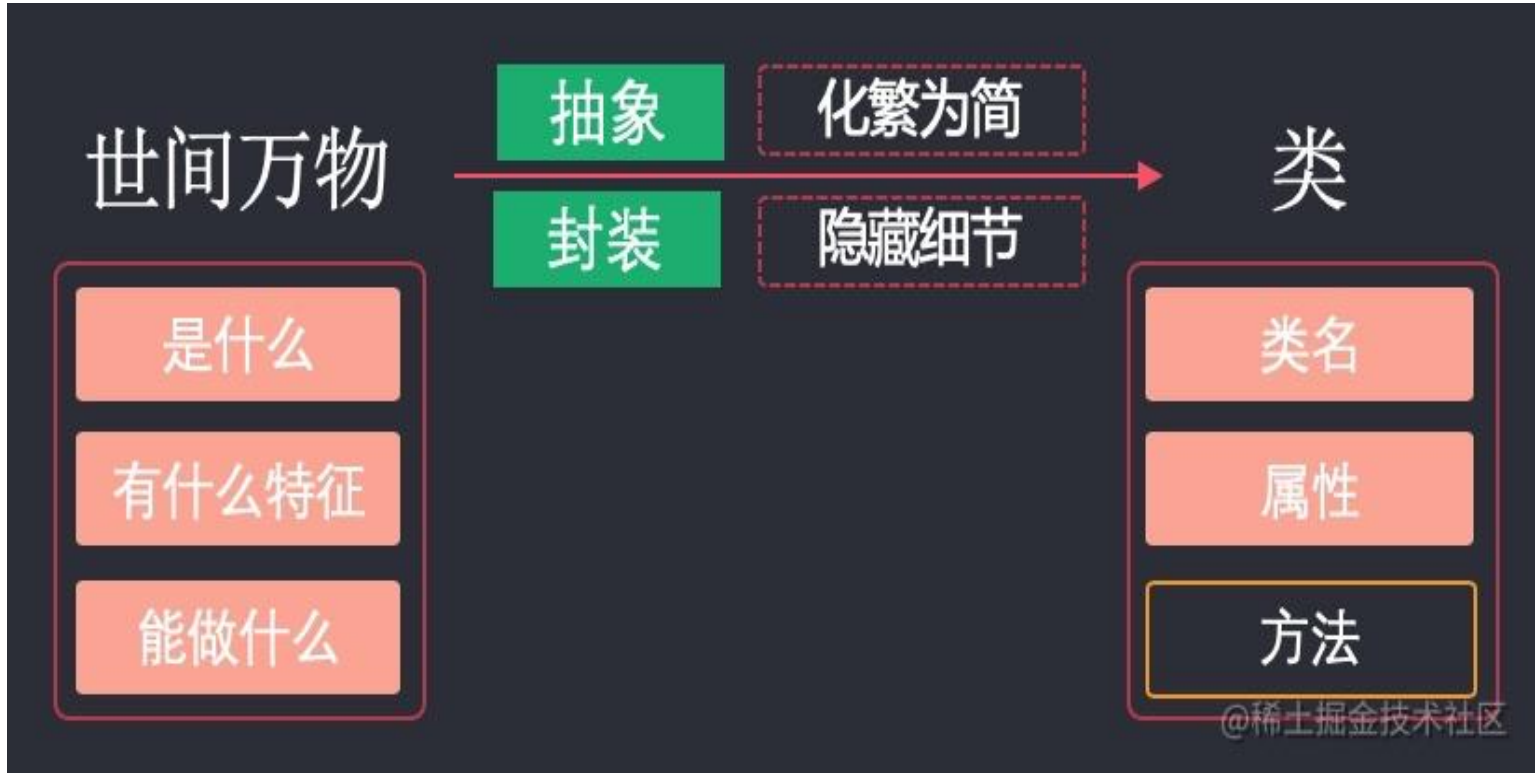




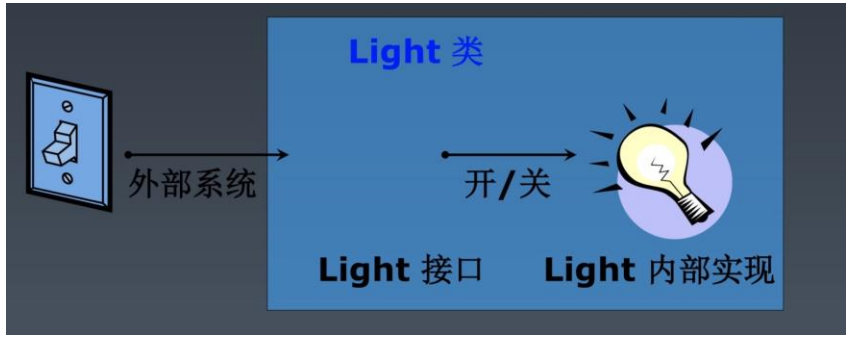
OOP： Object-Oriented Programming

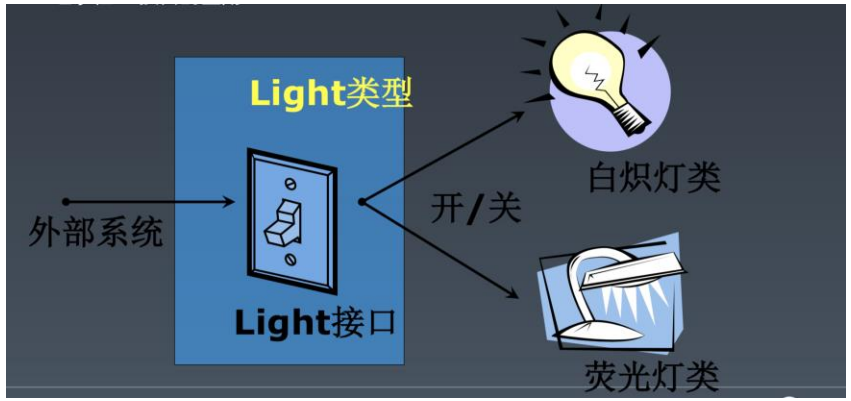
➢ 面向对象编程

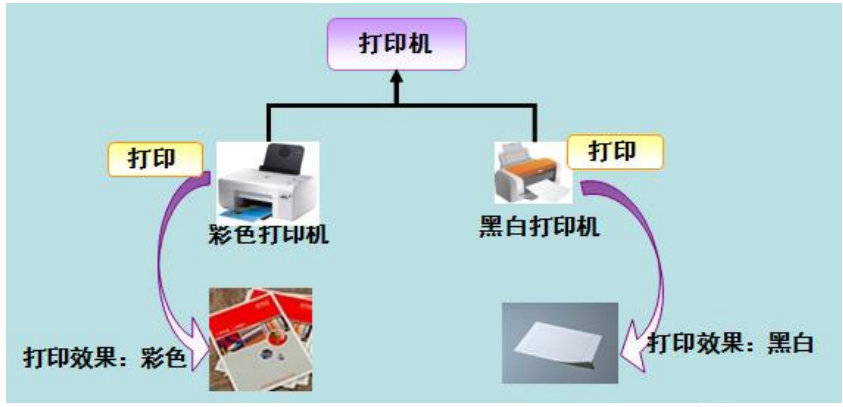
物以类聚，人以群分　万物皆对象，对象皆有类



https://juejin.cn/post/6844904041307176974



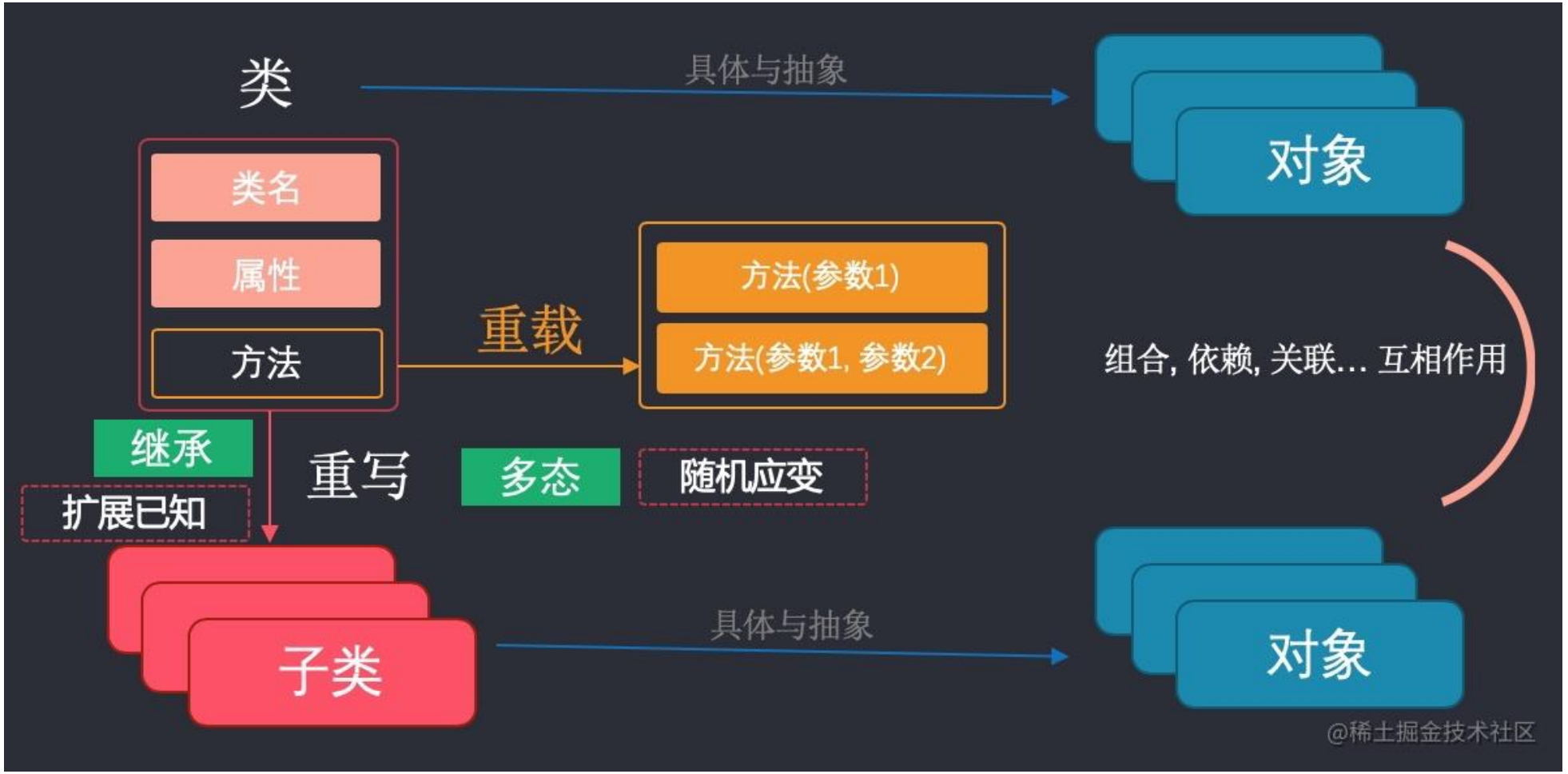




OOP： Object-Oriented Programming

➢ 面向对象编程

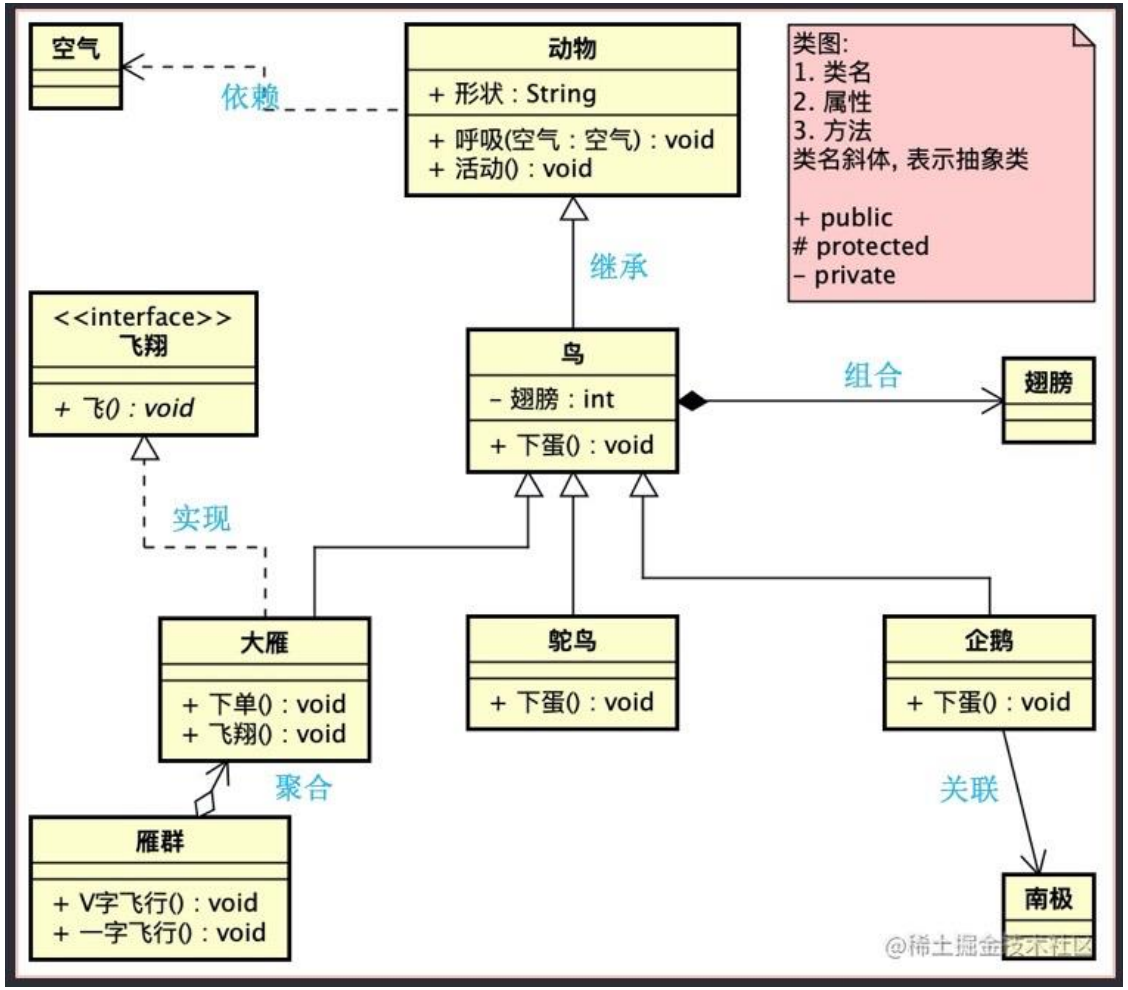
关系驱动世界



https://juejin.cn/post/6844904041307176974

OOP： Object-Oriented Programming

➢ 面向对象编程



https://juejin.cn/post/6844904041307176974

程序=算法+数据结构

软件=程序+文档

- **问题分析与描述**

  Problem Analysis and Specification

- **设计**

  Design

- **编码**

  Implementation (Coding)

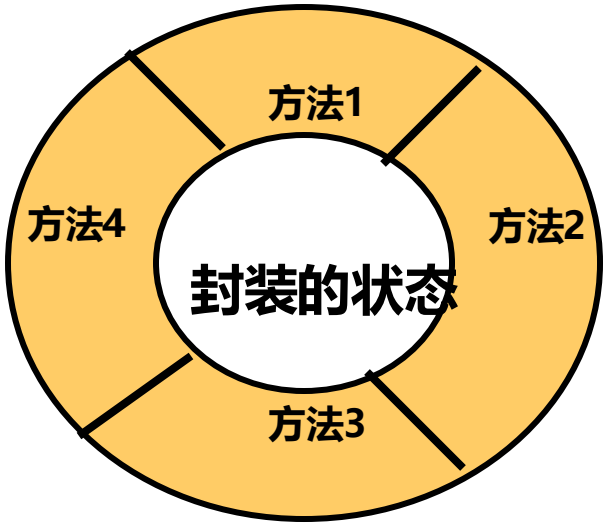
- **软件测试、调试**

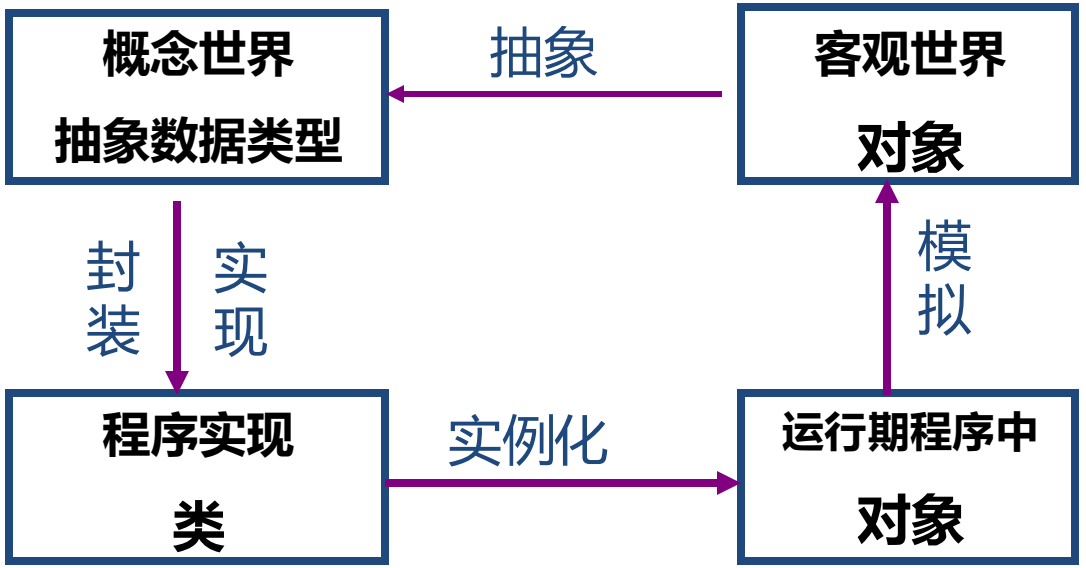
  Testing, Execution and Debugging

- **软件维护**

  Maintenance

以对象为中心的设计





- 1.1 从C到C++
- 1.2 最简单的C++程序
- 1.3 C++对C的扩充
- 1.4 C++程序的编写和实现
- 1.5 关于C++上机实践

2023年3月1日

1.2 最简单的C++程序?

回顾：最简单的C程序

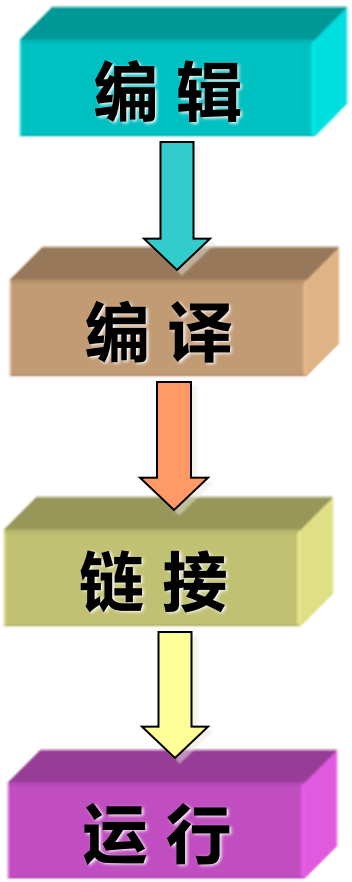
```
/* This is the first C program */
#include  <stdio.h>
int main ( )
{
   printf ("Hello World!\n");
   return 0;
}
```

注释信息
预处理命令
返回值为整型
无参数、返回值为整型的主函数
主函数名
函数开始
内部函数名
回车换行符
函数调用
参数
语句结束标志
函数返回值
函数结束
函数体

运行结果： Hello World!

1.2 最简单的C++程序?

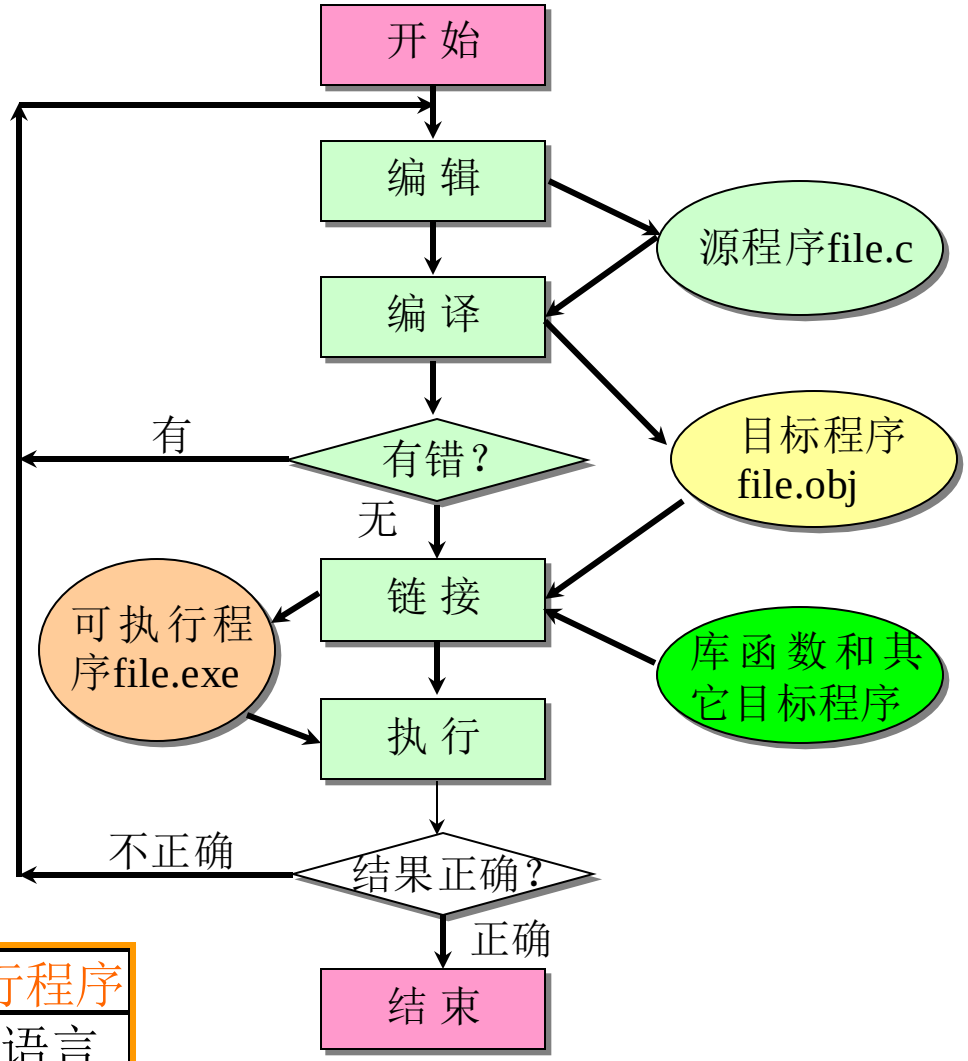
回顾：编程步骤



程序代码的录入，生成源程序*.c或*.cpp

语法分析查错，翻译生成目标程序*.obj

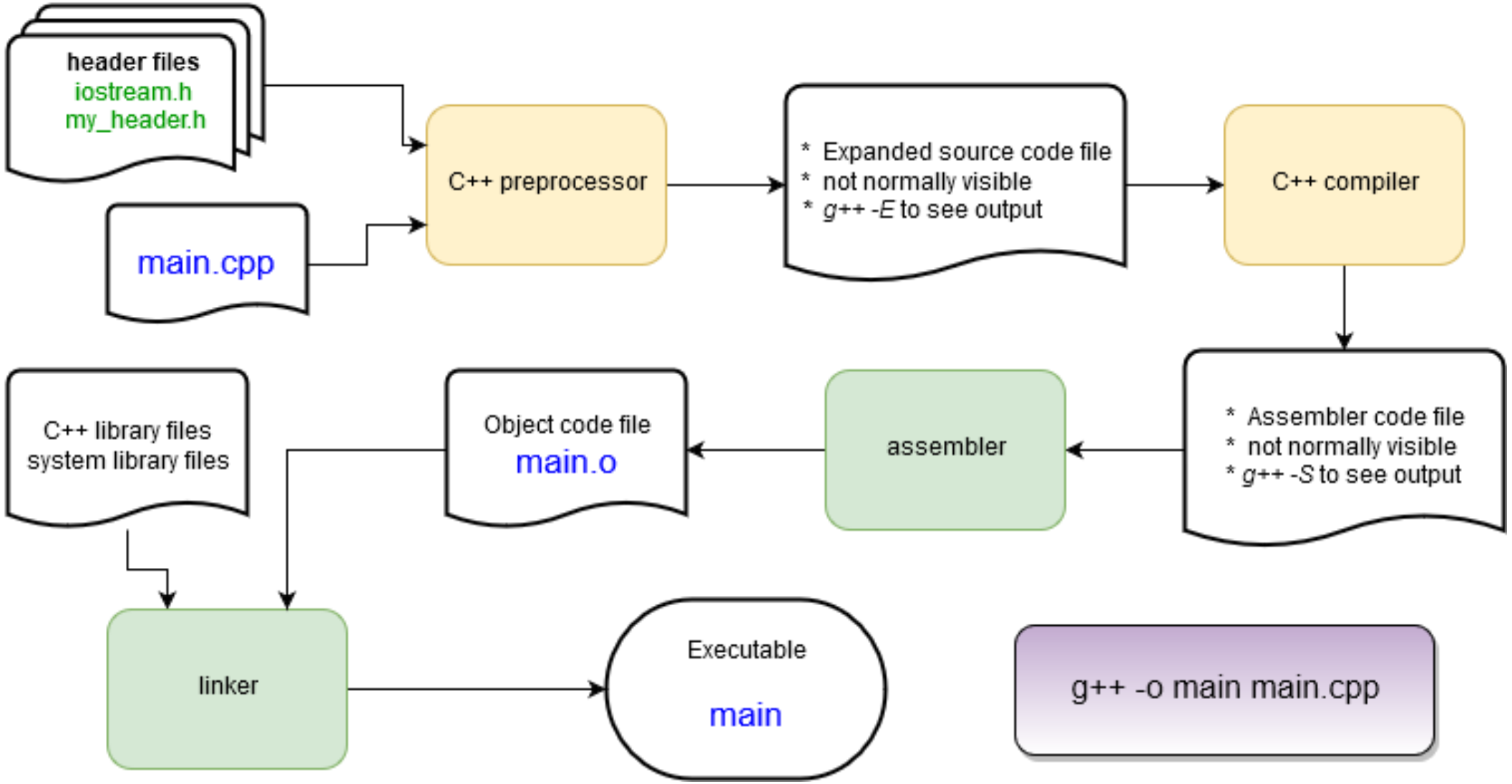
与其它目标程序或库链接装配,生成可执行程序*.exe



调试C/C++程序的流程

	源程序	目标程序	可执行程序
内容	程序设计语言	机器语言	机器语言
可执行	不可以	不可以	可以
文件名后缀	.c或.cpp	.obj	.exe

Behind the Scenes: The Compilation Process



1.2 最简单的C++程序?

例1.1 输出一行字符“Hello World！ This is a C++ program.”。

```
#include <iostream>   //用cout输出时需要用此头文件
using namespace std;  //使用命名空间std
int main()
{
  cout<<“Hello World!”;
  cout<<"This is a C++ program.\n"; //输出一行
  return 0;
}
```

```
#include <iostream>
using namespace std;
int main()
{
   cout<<"Hello world!"<<endl;
   return 0;
}
```

代码基本结构

输入输出方法?

程序的注释方法?

命名空间

程序运行过程

编程工具

什么是命名空间?

using namespace std;

C++新引入的一个机制，主要是为了解决多个模块间命名冲突的问题，就像现实生活中两个人重名一个道理。C++把相同的名字都放到不同的空间里，来防止名字的冲突。

```
#include<iostream>
using namespace std;
int main()
{
   cout<<"Nice to meet you!"<<endl;
   return 0;
}
```

```
#include<iostream>
int main()
{
   std::cout<<"Nice to meet you!"<<std::endl;
   return 0;
}
```

A note about STL naming conventions

- **STL** = Standard Template Library
  - Contains TONS of functionality (algorithms, containers, functions, iterators) some of which we will explore in this class

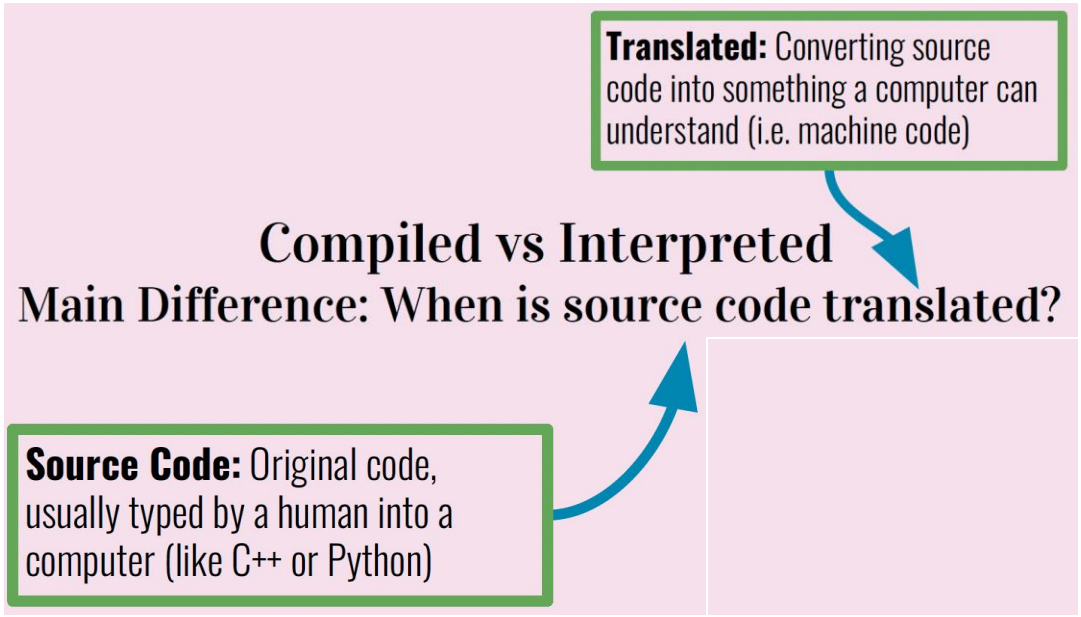
```
#include<iostream>
using  std::cout;
using  std::endl;
int main()
{
   cout<<"Nice to meet you!"<<endl;
   return 0;
}
```

C++新标准中使用不带.h的头文件包含时，必须要声明命名空间，并且包含头文件在前，声明使用的名字空间在后。

1.2 最简单的C++程序?

➢ Compiled, Interpreted, Dynamically/Statically typed

CS 106L



Compiled vs Interpreted: When is source code translated?

Dynamically typed, interpreted
- Types checked on the fly, during execution, line by line
- Example: Python

Statically typed, compiled
- Types before program runs during compilation
- Example: C++

Runtime: Period when program is executing commands (after compilation, if compiled)

static typing helps us to prevent errors before our code runs

C++ is a statically typed language

***statically typed**: everything with a name (variables, functions, etc) is given a type before runtime*

***dynamically typed**: everything with a name (variables, functions, etc) is given a type at runtime based on the thing's current value*

1.2 最简单的C++程序?

- **Types and Structs in C**

CS 106L

C++ Fundamental Types

```
int val = 5; //32 bits
char ch = 'F'; //8 bits (usually)
float decimalVal1 = 5.0; //32 bits (usually)
double decimalVal2 = 5.0; //64 bits (usually)
bool bVal = true; //1 bit

#include <string>
std::string str = "Sarah";
```

- **Types and Structs in C**

CS 106L

Fill in the types

```
string  a = "test";
double  b = 3.2 * 5 - 1;
int     c = 5 / 2;
int     d(int foo) { return foo / 2; }
double  e(double foo) { return foo / 2; }
int     f(double foo) { return int(foo / 2); }
void    g(double c) { std::cout << c << std::endl;}
```

1.2 最简单的C++程序?

- **Types and Structs in C**

CS 106L

```
struct Student {
string name; // these are called fields
string state; // separate these by semicolons
int age;
};
Student s;
s.name = "Sarah";
s.state = "CA";
s.age = 21; // use . to access fields
```

Definition

struct: a group of named variables *each with their own type.* A way to bundle different types together

1.2 最简单的C++程序?

例1.2　求a和b 两个数之和。

```
// 求两数之和(本行是注释行)
#include <iostream>//预处理指令
using namespace std;//使用命名空间std
int main()//主函数首部
{//函数体开始
  int a,b,sum;//定义变量
  cin>>a>>b;//输入变量a和b的值
  sum=a+b;//赋值语句
  cout<<"a+b="<<sum<<endl;//输出语句
  return 0;//如程序正常结束,向操作系统返回一个零值
}//函数结束
```

1.2 最简单的C++程序?

例1.3　求 a 和 b 两个数中的大数。

```
//例1.3 求两个数中的大数
#include <iostream>
using namespace std;
int main()
{
  int max(int x,int y) ; //对max函数作声明
  int a,b,c;
  cin>>a>>b;
  c=max(a,b); //调用max函数
  cout<<"max="<<c<<endl;
  return 0;
}
```

```
int max(int x, int y) //定义max函数
{
  int z;
  if(x>y) z=x;
  else z=y;
  return(z);
}
```

1.2 最简单的C++程序?

例1.4 包含类的C++程序。

```
//例1.4包含类的C++程序
#include <iostream>
using namespace std;
class Student // 声明一个类，类名为Student
{
private: // 以下为类中的私有部分
 int num; // 私有变量num
 int score; // 私有变量score

public: // 以下为类中公用部分
 void setdata() // 定义公用函数setdata
 { cin>>num; // 输入num的值
   cin>>score; } // 输入score的值

 void display() // 定义公用函数display
 { cout<<"num="<<num<<endl; // 输出num的值
   // 输出score的值
   cout<<"score="<<score<<endl; };
}; // 类的声明结束
```

```
Student stud1,stud2;
//定义stud1和stud2为Student类的变量，称为对象

int main() // 主函数首部
{
 // 调用对象stud1的setdata函数
 stud1.setdata();
 // 调用对象stud2的setdata函数
 stud2.setdata();
 // 调用对象stud1的display函数
 stud1.display();
 // 调用对象stud1的display函数
 stud2.display();
 return 0;
}
```

小结

1. 了解信息科学前沿方向和程序设计语言的流行度
2. 理解面向对象的基本思想
3. 掌握C++编程的基本步骤和编程环境使用方法
4. 掌握C++程序的基本特点
5. 初步理解包含类的C++程序

实验作业一 等额本金与等额本息还款计算器

因购房需要从银行贷款（程序可在运行时自行设定数额），贷款期限为10年（或15，20年，程序可在运行时自由设定），程序可在运行时自由设定贷款利率（请查询当前各大银行最新利率），请通过网络查询**等额本金**和**等额本息**两种**固定利率**还款方式的算法，**编写程序计算两种方式最终还款总额，并输出贷款年限内每月还款数额**。

要求：1.用C++语言设计类实现基本功能，功能正确，界面友好。

2.代码注释完整、清晰。

3.按要求的格式给出实验报告和总结。

4.两周内完成，在知新平台上传pdf版实验报告，并准备汇报PPT。

5.2025-03-10课上随机抽取同学进行PPT讲解和演示。

6.一旦发现雷同或抄袭痕迹，本次实验作业判零分。

2025-02-17

Thank You !

Q & A