



# 高级语言程序设计（进阶）

## 高级语言程序设计实践（进阶）

（国豪工科精英班）

任课教师：陈宇飞 同济大学电子与信息工程学院  
龚晓亮 同济大学电子与信息工程学院



# 1.1 教师简介

- 陈宇飞 副教授 博导
- 同济大学电子与信息工程学院
- 研究方向：机器学习，深度学习，医学影像分析
- 联系方式：

yufeichen@tongji.edu.cn

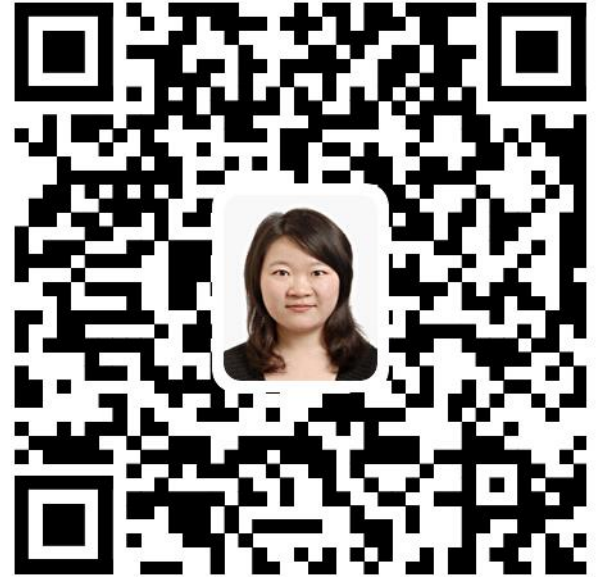




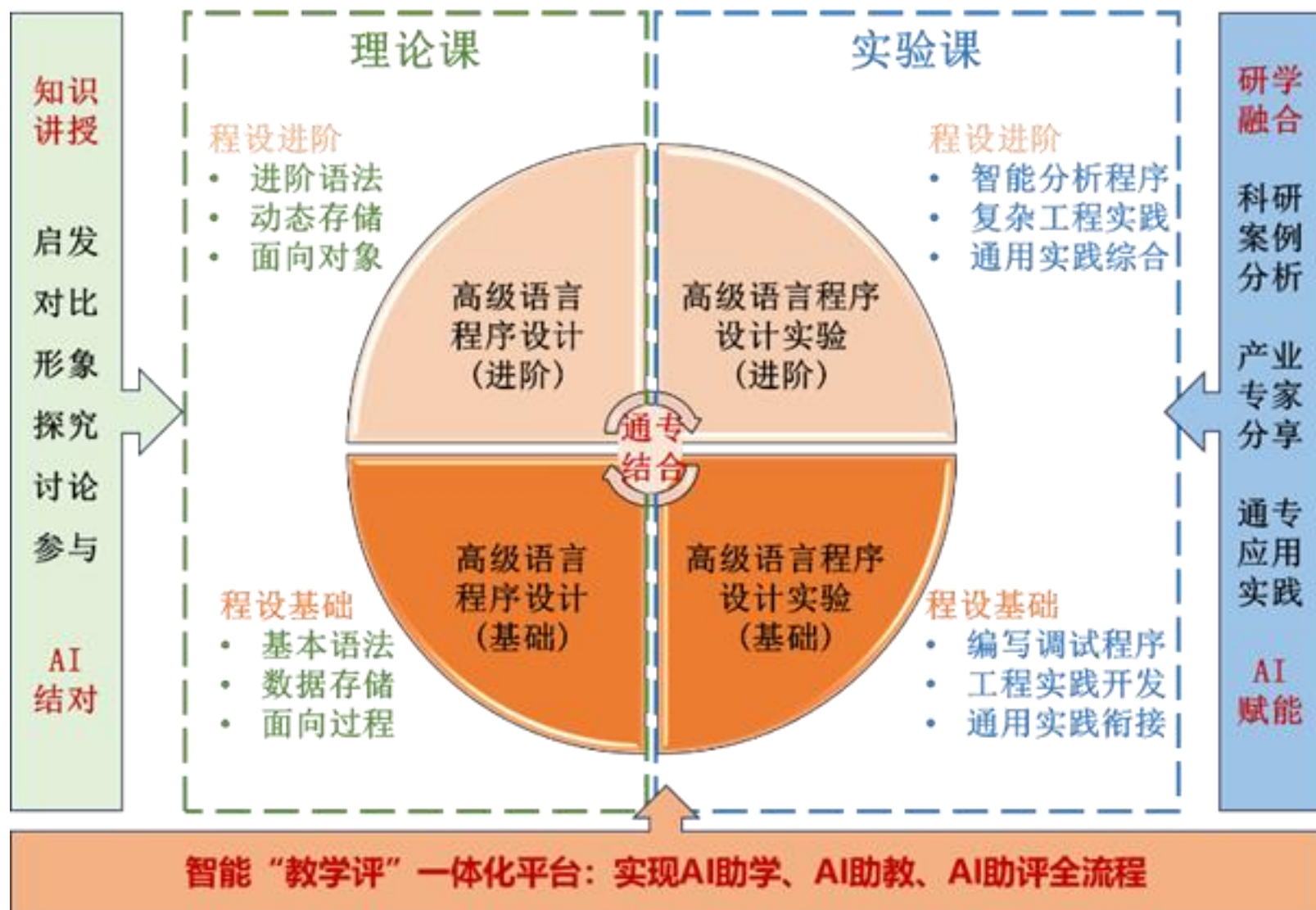
# 1.1 教师简介

- 龚晓亮 高级工程师 硕导
- 同济大学电子与信息工程学院
- 研究方向：情感计算、脑认知与智能计算
- 联系方式：

gxllshsh@tongji.edu.cn



## 1.2 背景简介



# 1.3 培养目标

- 区别于通识类程序设计课程：
  - **专**：培养学生专业的编程思维能力，使其能够完成专业系统的开发任务；
- 区别于信息类程序设计课程：
  - **通**：培养学生综合的编程能力，使其能够完成专题背景的综合开发任务。





# 1.4 课程特色

## ① 课程教学：

- AIGC驱动的程序设计
- 课程内容教研结合
- 含有AI研究课题相关进阶案例和实训
- OJ实时评测

## ② 实践教学：

- 主题研讨
- 朋辈互动
- 作业讨论
- 学术分享



# 1.4 课程特色

- 高程进阶（国豪工科精英班）：
  - 高程进阶章节作业
  - 高程进阶大作业（3大类，每类选择1个完成）
    - 设计类：创意时钟、拓扑造型图、创意字符画
    - 算法类：文本压缩、图像压缩、理论配筋
    - 应用类：贪吃蛇、医学影像分析、海藻推演
  - 高程进阶期末作业（商飞题目：虚拟发动机状态评估模拟器）



# 1.4 课程特色

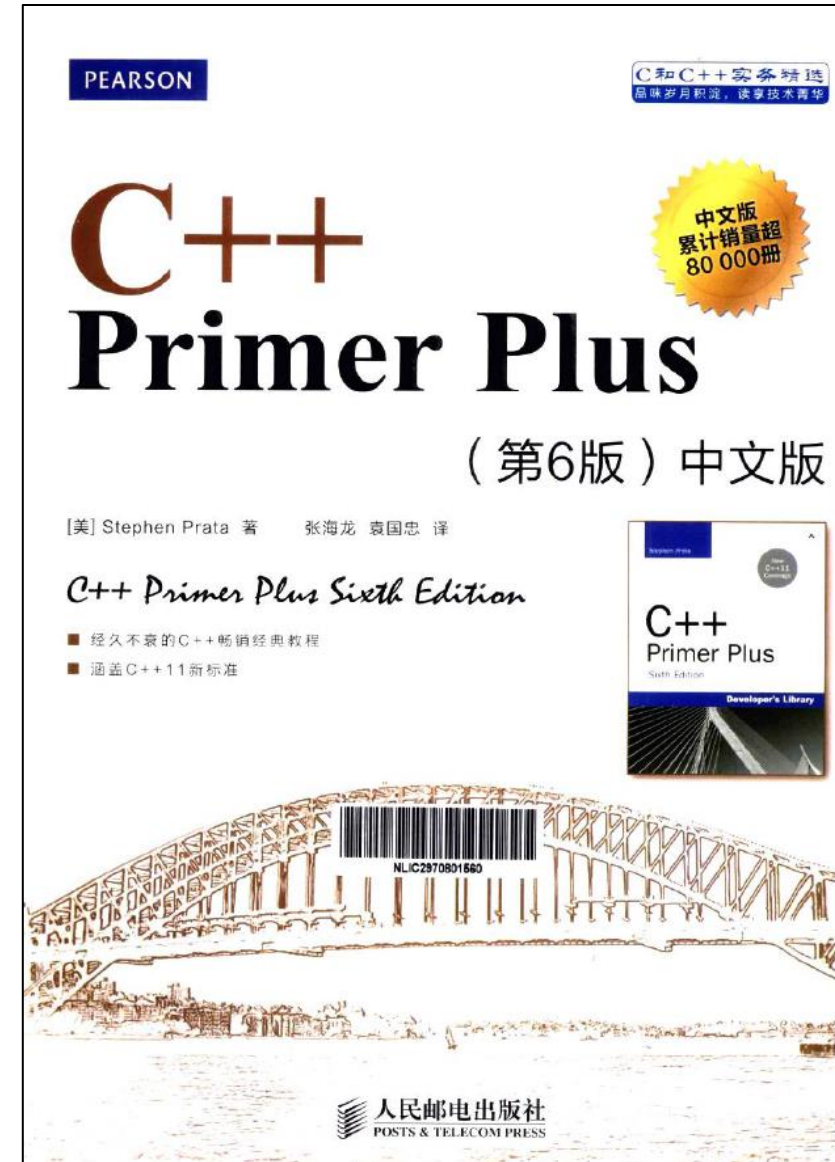
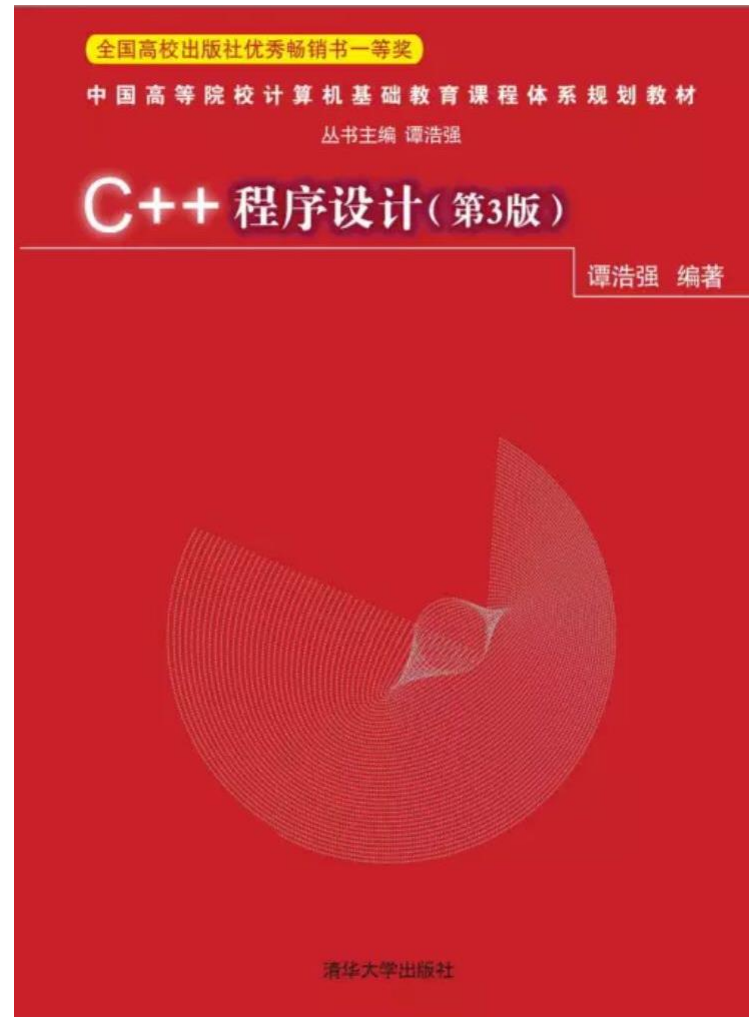
- 高程进阶（国豪工科精英班）：同比上学期基础训练
    - 高程进阶章节作业：适当降低频率和数量
    - 高程进阶大作业（3大类，每类选择1个完成）
      - 设计类：创意时钟、拓扑造型图、创意字符画
      - 算法类：文本压缩、图像压缩、理论配筋
      - 应用类：贪吃蛇、医学影像分析、海藻推演
    - 高程进阶期末作业（商飞题目：虚拟发动机状态评估模拟器）：校企联动
- 面向进阶应用  
强调个性化实战  
力争题目一次性放出  
取消普通大作业报告





# 1.5 课程内容

## ● 教材说明





# 1.5 课程内容

周次	课程内容	
1	高级语言程序设计进阶	(1)开学篇：课程目标、内容、考核要求、答疑方式等； (2)补充概念：共用体、位运算、带参数的main函数； <b>(3)智能编程与OOP设计。</b>
2	动态内存申请与释放	(1)动态内存的基本概念； (2)C中的相关函数； (3)C++中的相关运算符； (4)内存的动态申请与释放； (5)定位new运算符。
3	链表的实现与应用	(1)链式结构的基本概念； (2)单链表的基本操作； (3)双向链表的基本操作； (4)二叉搜索树。
4	包含动态内存申请的类	(1)含动态内存申请的构造函数和析构函数； (2)构造函数与析构函数的调用顺序； (3)对象的动态建立和释放； (4)对象的赋值与复制。
5	C方式的文件操作	(1)文件概述； (2)文件的打开与关闭； (3)文件的读写； (4)其他标准I/O函数。
6	C++方式的文件操作	(1)标准输入流； (2)文件操作与文件流； (3)字符串流。
7	类和对象的基本使用	(1)对象的初始化方法； (2)构造函数的基本概念与使用； (3)析构函数的基本概念与使用； (4)复制构造函数的调用时机。
8	数据共享与参数化类	(1)共用数据的保护； (2)静态成员； (3)类模板。
9	运算符重载的基本概念	(1)运算符重载的方法； (2)运算符重载的规则； (3)成员函数和友元函数。

10	运算符重载的实例应用	(1)单/双目运算符的重载； (2)流运算符的重载； (3)不同类型间数据的转换。
11	继承与派生的基本概念	(1)继承与派生的基本概念； (2)派生类的声明方式及派生类对象的构成； (3)派生类的成员访问属性； (4)派生类的构造与析构函数。
12	多重继承的概念与实现	(1)多重继承； (2)虚基类； (3)赋值兼容规则； (4)继承和动态内存分配。
13	多态性与虚函数	(1)多态的基本概念； (2)虚函数的定义和使用； (3)虚析构函数； (4)纯虚函数与抽象类。
14	模板类	(1)智能指针模板类； (2)标准模板库。
15	泛型编程	(1)STL基本概念； (2)使用迭代器； (3)迭代器类型； (4)迭代器结构； (5)容器种类。
16	跨平台编译	(1)Cmake编译原理； (2)Makefile的使用。
17	复习答疑	



# 1.5 课程内容

- 关于作业

- 全部在线提交，提供OJ实时评测：<http://202.120.167.244/>
- 编译环境：VS 2022；VS+Copilot
- 认真阅读帮助中心的使用手册
- 评测得分以最后一次为准
- 无客观理由，不接受截止后的作业补交
- **禁止抄袭**，全部题目全部代码均查重！！！！
- 认真读题，读要求！！！！



4_3/ <span style="background-color: #008000; color: black;">          </span> .cpp (74%)	<span style="background-color: #ff0000; color: black;">          </span>	4_3/ <span style="background-color: #008000; color: black;">          </span> .cpp (74%)	<span style="background-color: #ff0000; color: black;">          </span>
<a href="#">7-27</a>	<span style="background-color: #ff0000; color: black;">          </span>	<a href="#">16-36</a>	<span style="background-color: #ff0000; color: black;">          </span>
<a href="#">32-39</a>	<span style="background-color: #008000; color: black;">          </span>	<a href="#">6-13</a>	<span style="background-color: #008000; color: black;">          </span>



4\_3/            .cpp

```
#include<iostream>
#include<cmath>
using namespace std;
double getDouble();
double solve(double a, double b, double c, double d);

double getDouble()
{
    double x;

    cin >> x;
    if (!cin.good() || x < -1e9 || x > 1e9)
    {
        cout << "Input is illegal, program exit.";
        exit(0);
    }

    else return x;
}

double solve(double a, double b, double c, double d)
{
    double x, x0, f1, f2;
    for (x = 1, x0 = 0; fabs(x - x0) > 1e-7;)
    {
        x0 = x;
        f1 = ((a * x0 + b) * x0 + c) * x0 + d;
        f2 = (3 * a * x0 + 2 * b) * x0 + c;
        x = x0 - f1 / f2;
    }
    return x;
}

int main()
{
    double a, b, c, d;
    a = getDouble();
    b = getDouble();
    c = getDouble();
    d = getDouble();
    cout << solve(a, b, c, d);
    return 0;
}
```

4\_3/            .cpp

```
#include<iostream>
#include<cmath>
using namespace std;
double getDouble();
double solve(double a, double b, double c, double d);

int main()
{
    double a, b, c, d;
    a = getDouble();
    b = getDouble();
    c = getDouble();
    d = getDouble();
    cout << solve(a, b, c, d);
    return 0;
}

double getDouble()
{
    double x;

    cin >> x;
    if (!cin.good() || x < -1e9 || x > 1e9)
    {
        cout << "Input is illegal, program exit.";
        exit(0);
    }

    else return x;
}

double solve(double a, double b, double c, double d)
{
    double x, x0, f1, f2;
    for (x = 1, x0 = 0; fabs(x - x0) > 1e-7;)
    {
        x0 = x;
        f1 = ((a * x0 + b) * x0 + c) * x0 + d;
        f2 = (3 * a * x0 + 2 * b) * x0 + c;
        x = x0 - f1 / f2;
    }
    return x;
}
```



# 1.5 课程内容

- 关于实践课
  - 不安排机房上机，实践课同理论教室
  - 自行配置好编译环境和工具
  - 主题讨论与辩论
  - 邀请专家/同侪/朋辈报告
  - 大作业交流讨论（更多的应用问题）



# 1.6 考核方法

- 理论课

**成绩1** – 作业完成情况（不安排期中考试）（作业=章节作业+大作业）

**成绩2** – 期末考试（大作业+汇报展示）

- 平时成绩、期末成绩分开计算，满分均为100

**总评成绩** = 成绩1 x **70%** + 成绩2 x **30%**

=> 百分制转五分制

- 若遇特殊情况，以新通知为准



# 1.6 考核方法

- 实践课

成绩1 – 作业完成情况 (作业=章节作业+大作业+互动讨论)

成绩2 – 实验报告成绩 (大作业需要撰写实验报告)

- 成绩1、成绩2的分数单独计算，满分按100折算

总评成绩 = 成绩1 x 70% + 成绩2 x 30%

=> 百分制转五分制





# 1.7 课程通知

- QQ群：
  - 不鼓励水群
  - 不鼓励开临时对话实时提问
- 知识点疑问课后答疑（45分钟/周）
- 正式的课程联系方式
  - **OJ**: 作业网站、交流讨论
  - **Canvas**: 课程资料、重要通知



扫一扫二维码，入群聊







# 1.8 课程建议

- 章节作业:

- 理论课知识点
- 题面描述直观

- 大作业:

- 难点: 题面分析 + 设计
- 需要自学 (课外延展) + 研讨 (鼓励交流)

# 1.9 课前问卷

- 做个问卷调查吧~~~

高级语言程序设计进阶-国豪-课  
前调查-2024



长按识别二维码