黄永峰 62792710 [yfhuang@tsinghua.edu.cn](mailto:yfhuang@tsinghua.edu.cn)

办公室：罗姆楼5-104（电子系信息认知与智能系统研究所）

严谨思维训练、（由于学时数少）基本方法训练（而不局限于C）

课程宗旨：

自信自强（坚持学下去就能学好） 协作共赢（大程序布置一起开发）

大国工匠（低级技术活反复练习） 人类价值

&

严谨态度（计算机很傻自主性差） 创新思维（算法有创新空间 多方法完成基本要求）

真知实践（经验感受来自实践 电脑完成作业） 顶层设计（算法自顶逐步细分）

知识体系：（学习内容三方面）

编程（做菜）=算法（菜谱）+数据类型（食材）+*工具*（锅）； **程序=算法+数据结构/类型**

C语言编程，思路，平台使用（玩游戏的心态，不是重点） 的相关知识

教学目标：

C→C++ 面向过程程序设计的基本知识**方法**（核心部分，C和C++差别不大，包含关系）

本学期C（子集），下学期C++（全集）

其它语言的方法也类似，学起来也容易 （java python自学）

基本编程和**调试能力**（反复练习 发现错误 编程不可能不出错 计算机难以读懂错误）

调试能力需要练好，各个语言类似，发现问题快则效率高，只能自己练出经验，学会就不忘

算法是永无止境的 短期搞不定

养成良好编程素养（而不只是能力）：

1. 强调代码可读性好（注释、对齐）（第三方也能读懂 而且大程序需要）

[不影响运行但是显示水平]

2. 程序构思有说明（设计文档写清晰思路）

[需求调研→需求文档→概要设计文档→**详细**设计文档（详细到看着就能直接写代码）→代码开发（码农 最低级 最累）→测试阶段（占30%工作量 测试各种可能）]

3. 学会寻找BUG（调试 三倍的时长）

4. 算法和结构尽可能更好（优化）（与处理领域专业知识有关 最终最难的部分）

5. 运行结果正确性分析（测试）（反复尽可能完备 有限场景覆盖尽可能多的情况）

[期中期末两个机考 需要注意测试正确性尽可能完备 多跑几个数据]

重在方法学习，自己进行的技能训练，鼓励创新

学习方法：

老师传道+学生练术 **不是懂不懂而是写不写得出来**

（生活中的问题电脑都能解决，只是算法不知道）

自学做题（电脑上）+上机应用（实际意义的问题）+*上网*交流（细节差异问题）

码农素质：耐烦 耐劳 （电脑 反复不通 直到调试能力提高后会好一点）

考核方式：

实践——实验作业（电脑上的代码、测试结果等）30%+期中机考10%+期末机考20%（当场做当场批改）

笔试（核心课 闭卷考试）——期末笔试40%（还是看编程能力 曾经考阅读能力 今年有变）

[曾经的试题系里今年全部公开 所以有变数]

网络讨论约3% （成绩微调的余量 创新创意等网络上交流可以调分）**[网络学堂板块 活跃]**

作业提交：电子版（网络学堂）第一次上机问助教细节

注意作业提交时间和截止时间（逾期的话邮件发给老师/助教，***态度***好不扣分，比不发好）

网络学堂留下电话等联系信息

讨论方式：网络学堂/email/答疑（周五全天在办公室，其他天先打电话）

挂科很难，查重出问题就容易了，免费作业查重！（C语言很难一模一样）

网络学堂“课堂讨论”板块，老师助教都在 弥补上课不足 问问题/回复越多越好 积极参与

（把网络学堂黑掉老师支持）

上机不考勤，自己有电脑也不要求去，只看结果（与地点无关）

机房有助教可以手把手教 建议初学去 主楼9楼东边微机实验室

机房vs2012/2018编译工具（考试用） 建议mac/linux的编译工具也都试试

不同版本vs差别不很大 助教5个

组长王慧丽 62792516/18813167835 可以以此询问其他助教消息

参考教材第一章

计算机基础（高中有的讲了所以不开课）

第一台数字电子计算机ENIAC（冯诺依曼体系：二进制；程序顺序执行[不是存储顺序]）

基本组成：

显示器 主机等→硬件；

软件（更有价值）[操作系统（管理硬件 基础 三大 国内在做）；编译器（平台）；应用软件]

核心部件：

运算器、控制器→CPU（代码的执行和控制）；

（内）存储器（输入保存、CPU提取，电保存断电后删除）

硬盘（外部存储，不工作时保存仓库，磁保存[磁粉极性保存]，很难打开，需要“存盘”操作）

输入设备（键盘 鼠标等）、输出设备（显示器等）

编程大概过程：

问题→制定算法（思路）→翻译为C语言等→输入（保存到存储器）→编译（把高级语言翻译成01编码机器码）→执行→输出/程序保存到硬盘

程序是若干条指令的有序集合（**程序结构**）

程序按顺序存储在内存（内部存储器）中（**存储类型**）

CPU按程序结构执行指令（而非存储顺序）（**算法过程**）

程序都有输入输出操作（**数据1/0**）

基本能力：

制定算法（分析问题）；

编辑代码（编程素养）；

编译链接（调试能力——编程错误；计算机的报错和人的理解）；

执行调试（测试能力——算法错误）。

程序设计基础：

语言——C开发工具VS20xx；语法词法函数

数据结构；算法

编程语言：（和硬件结合度递降，发展时间递近，能力递降）

机器语言（01，现在一般不学，开发CPU使用，最厉害😊）

汇编语言（助记符表达机器语言，与特定计算机有关，硬件设计使用**[数字逻辑]**）

高级语言（人易于理解的类自然语言，通用，应用程序使用）

C与编译器有关；JAVA与编译器无关（更高级，自学容易）

C语言：中级语言（在汇编和高级之间），操作系统用的，适用面广，教学语言

应用程序一般python/java

算法：解决一个实际问题采取的步骤和方法

数值算法——数学方法、数值表示 数学建模（没有难点了，有不同算法但有最优算法）

[必须有规律，步骤必须有限！]

非数值算法——没有固定模式（如做菜，难！）

算法描述：

自然语言（通俗但是冗长且可能歧义）；

流程图描述法（基本框图符号）（⚪分页防止图过大）

[圆矩形——起止框；平行四边形——输入输出框；菱形——判断框；矩形——加工框]

分支判断；循环结构 都可以写

写程序之前，最好先画一个流程图，就很清楚

C语言70年代诞生

基本结构：（先照葫芦画瓢[如前四行]至期中考试）

C程序→源程序文件（许多）→预编译部分；函数[主；非主]→说明部分[名称]；执行部分

多个文件组成，一个程序有且只有一个主函数main()，程序从main()开始执行和结束

函数由C语句组成（语句是基本单位，每句后面有一个分号）

对变量，先定义数据类型（设为变量才能用），再使用

数据的输入输出printf()、putchar()和scanf()实现（先照葫芦画瓢），每个程序都需要有

//（C++）或/\*…\*/（C）进行注释，每句话注释

数学函数可以直接调用；10函数

System(“pause”)屏幕暂停语句

上机方法：（助教手把手教）

程序开发四个步骤：※※※

键盘**编辑**cpp/c→磁盘**编译**（按键）obj（以文件为单位）

→**连接**（编译器也给做了）exe（几个文件合一+标准函数）→**执行**（也不一定符合要求）

编译提示不一定是对的（如第三行报错实际可能在第二行有问题）

调试练习：自己把对的改错收集报错信息

必做题（非做不可）和选做题（扣分补回来）