



# 数据结构

## Data Structure

任课老师：刘利刚

[ligangliu@zju.edu.cn](mailto:ligangliu@zju.edu.cn)

[http://www.math.zju.edu.cn/ligangliu/Courses  
/DataStructure\\_2005-2006](http://www.math.zju.edu.cn/ligangliu/Courses/DataStructure_2005-2006)

A decorative graphic on the left side of the slide, consisting of overlapping blue, red, and yellow squares with a black crosshair.

# 讨论、经验分享

---



# 竺可桢校长问的两个问题

- 诸位在校，有两个问题要自己问问：一是你来浙大干什么？二是你毕业后要做一个什么样的人？
- 认真思考？or 茫然？



# 工作中的反思： 我的二十年教育的缺憾

- 在学校学的东西大部分对我后来的工作没有直接帮助
- 我在工作中碰的大部分问题在学校没有学到
- 如果我能再回到学生时代 ...
- 回到学校
  - 继续学习该学的东西
  - 做自己感兴趣的事情
  - 更多地帮助我的学生



# 大学期间须培养 and 发展的能力

- 正确的世界观，人生观
  - 积极的态度，接受挑战
  - 热爱生活，热爱专业，热爱工作
  - 认识自我，合理定位
- 个人素质
  - 良好的身体素质：“身体是革命的本钱”
  - 健全的心理素质：品德，个性
- 个人能力
  - 学习（自学）能力
  - 专业知识与技能
  - 创造性思维
  - 发现，分析和解决问题的能力，表达能力，交流能力等



# I. 做一个正直的人

- 伟大的人物都有伟大的人格
- 加强个人修养：做事先会做人
  - 坚强意志高尚品德
  - 情绪，情感，气质
  - 谦让，克己，宽容，谨慎，负责...
- 自我意识：挑战自我，开发自身潜力
  - 自我改造，自我监督，自我完善
- 生活在群体之中（团队精神）



## II. 对生活和工作充满激情

- 爱自己，爱自己的家人
- 爱因斯坦：“在学校和生活中，最重要的动机是乐趣，是工作中获得结果的乐趣，以及对这个工作的社会价值的认识”
- 做一个主动的人
  - “态度决定一切”
  - 全身心投入，忘我工作
- 积极心态，积极进取
  - 战胜挫折



# III. 培养较强的自学能力

- 养成思考的习惯
- 扎实的知识
  - 数学
  - 英语
  - 计算机
  - 专业知识
- 动手能力
- 创新能力
- 学无止境
  - 虚心求学：“三人行，必有我师”





# 回答竺校长的问题

- 你来浙大做什么？
  - 我来浙大获得一种思想的启蒙
  - 我来浙大获取知识，培养自身的能力
- 毕业后做一个什么样的人？
  - 做一个求是的人，求就是探索，是就是本质
  - 人的内心世界与外部世界的动态关系的一生的自觉探索过程—对人生的更多思考



# 千里之行，始于足下

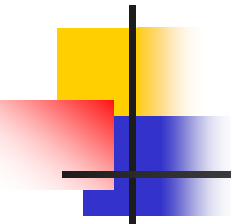
- 人生的目的
  - 使自己的“影响力”最大化
    - 使自己，自己的亲人生活得更好
    - 使自己周围的人生活的更好
    - ...
- 大学的目的
  - 学习和培养最大化自己影响力的能力



# 珍惜大学最黄金的好时光！

- 自信
  - 相信我能做得最好
- 努力
  - 只要勇于付出，未来将是灿烂一片
- 乐观
  - 不经历风雨，怎能见彩虹



- 
- 
- 做自己感兴趣的事情
  - 成功等于99%的汗水和1%的灵感！
  - 自信，快乐，成功



# 大学生的境界

---

- 不学
- 教什么学什么
- 喜欢什么学什么
- 有目的地学



# 学习的境界


- 学会答案
  - 学会一个解
- 学会方法
  - 学会一个方法
- 学会学习
  - 学会找到方法的方法
- 学会做事
  - 学会超越“纸上谈兵”，与实际“融会贯通”



# 生存的境界

- 改变你能改变的
- 有度量接受你不能改变的
- 有智慧区分能改变的和不能改变的
- 改变我们自己
  - 在现有的环境下，最大限度的实现自己的理想



- 
- A decorative graphic in the top-left corner consisting of overlapping yellow, red, and blue squares with a black crosshair.
- 
- 任何时候开始努力都是来得及的！
    - “再这样下去，我的一生就毁了。”





# 讨论

- 各位目标
  - 学好？
    - 考试高分
    - 编程高手
  - 混学分？
    - 应付考试
- 希望的技能
  - C or C++?
  - Turbo C/C++ or Visual C/C++?
- 授之以鱼，不如授之以渔
- Enjoy coding 😊



# Course Goals

- 让同学们真正能
  - 掌握编程语言
  - 设计合理的数据结构
  - 实现自己的算法
- 方法
  - 实践
  - 实践
  - 实践 Practice makes perfect!



# Why C++?

- 面向对象编程语言
  - 代码重用
- 特性
  - 封装
  - 继承
  - 多态



# Why Visual C++?

- 提供很高的编程能力和方便性
- 内容
  - VC++ Developer Studio
  - VC++ Runtime Libraries
  - Microsoft Foundation Class (MFC)
    - 扩展C++类库，用于生成Windows GUI程序



# 编程是一门艺术

- 一流代码的特性
  - 鲁棒 - Solid and Robust Code
  - 简洁 - Maintainable and Simple Code
  - 高效 - Fast Code
  - 简短 - Small Code
  - 共享 - Re-usable Code
  - 可测试 - Testable Code
  - 可移植 - Portable Code
- 多实践，多学习别人的代码，学无止境！



# 写好代码的一些技巧（1）

- 遵循一定的编程规范

- 注释
- 尽量用‘{‘和’}’，单独占行且对齐 Ex
- 取个好名字

- 预警机制

- 变量初始化
- 检查返回值
- 避免计算无效



# 写好代码的一些技巧（2）

- 处理任何可能出错的地方
  - 不要假设它们不会出现，它们总是会出现
  - Out of memory
  - Thrown exception
- 充分测试
  - 利用debugger
  - 不要“假设”
  - Code review



# Visual C++入门

---

- Win32 Console Application
  - 不必理会Windows GUI编程的复杂性
- MFC GUI Application
  - 轻松设计Windows GUI程序





# 指针

---

- A difficult point in C/C++



# 内存分配与释放

---

- 用 new 和 delete
  - 不要用 malloc 和 free



# new 和 delete

- 调用new所包含的动作
  - 从系统堆中申请恰当的一块内存
  - 若是对象，调用相应类的构造函数，并以刚申请的内存地址作为this参数
- 调用delete所包含的动作
  - 若是对象，调用相应类的析构函数
  - 将该内存块返回给系统堆



# new[] 和 delete[]

- 调用new[]所包含的动作
  - 从系统堆中申请可容纳n个对象外加一个整型的一块连续内存
  - 将n记录在额外的那个整型内存中
  - 调用n次构造函数初始化这块内存中的n个连续对象
- 调用delete[]所包含的动作
  - 从new[]记录n的地方将n值找出
  - 调用n次析构函数析构这块内存中的n个连续对象
  - 将这一整块内存(包括记录n的整型)归还系统堆



- Note: 释放后指针应赋予一个合理的值
- For example:

```
int *pData;
```

```
...
```

```
delete pData;
```

```
pData = NULL;
```

A decorative graphic consisting of overlapping yellow, red, and blue squares with a black crosshair.

# C $\rightarrow$ C++ (1)

---

- An Example

```
int n;  
int *pData;
```

- An Example

```
typedef struct  
{  
    int n;  
    int *pData;  
}NVector;
```

A decorative graphic consisting of overlapping yellow, red, and blue squares with a black crosshair.

# C $\rightarrow$ C++ (2)

---

- An Example

class NVector

{

    int n;

    int \*pData;

};



# C → C++ (3)

```
template <class Type>
class dataList
{
    int n;
    Type *pData;
};
```

template: 适合多种数据类型的类定义或算法，  
在特定环境下通过简单地代换，变成针对具体  
某种数据类型的类定义或算法



A decorative graphic in the top-left corner consisting of overlapping yellow, red, and blue squares with a black crosshair.

---

- C++ is Easy !



# Grading Policies

---

- General homework (30%)
  - Every class/week
- Large projects (40%)
  - 2-3
- Final exams (30%)



# Grading Policies

5+	算法实现新颖巧妙，非常好
5	算法实现正确，代码写得很好
4	算法实现正确，代码写得好
3	算法实现基本正确，但代码写得一般
2	算法实现有问题，代码一般
1	算法实现不正确，代码糟糕，态度有问题
0	态度有问题，很遗憾 ☹️

对你的作业程序的反馈会返回给你。



# Requirements(1)

## ■ 代码编写

- 符合一定的规范
- 代码清晰
- 尽量多的注释

## ■ 代码提交

- 文档说明：运行参数，操作等
- 程序能运行：编译通过
- 压缩打包
  - 去除不必要的文件，如\debug目录
  - Filename: yourID\_Homework\_#.zip    yourID\_Project\_#.zip



# Requirements(2)

## ■ File Headers

- `/*****\`
- Department of Mathematics, Zhejiang University
- Copyright (c) 2004 Department of Mathematics, Zhejiang University
  
- Author:
- Your ID, name and other information
  
- Module Name:
- An unabbreviated name for the module (not the filename)
  
- Abstract:
- Description of what this module does
  
- Notes:
- [Optional] Additional notes about this module - things that may help the reader of this code later on. Examples are: algorithm description, special case conditions, references, etc.
  
- History:
- Created on mm/dd/yyyy by email-name
- Modified on mm/dd/yyyy by email-name
- [Optional] history description
- `\*****/`



## ■ Function headers

- `/******\`
- Function Description:
- Description of what the function does
- Arguments:
- `[<blank> | OUT | IN/OUT] argument-name - description of argument`
- `...`
- Return Value:
- `return-value - description of return value`
- `or NONE`
- History:
- `Created on mm/dd/yyyy by email-name`
- `\*****/`



# Requirements(3)

- Input and Output
  - Your program must read from a file “**input.txt**” (if there is any input) and write to a file “**output.txt**” in the *current directory*.
- Judge compiler
  - Visual C++ 6.0 Enterprise Edition
  - or: Visual .net 2003



# Requirement(4)

---

- Assignment submissions
  - No late assignments will be accepted
  - Late assignment follow the following rules
    - 25% deduction for 1-day late
    - 50% deduction for 2-day late
    - Not accepted after being 2-day late
  - Feedbacks on your codes will be returned to you after judging





# Requirement(5)

---

- 独立完成
- 相互帮助
- 团队合作
- 绝不允许抄袭！



# 上机时间

---

- 每周一下午9 , 10节3:30-
- 紫金港中心机房1楼1号机房60-90座位
- 上机答疑



# Assignment

- A Vector Class
  - 向量之间加减，数乘等
- A Matrix Class
  - 矩阵之间加减，数乘，矩阵转置等
- Requirement
  - 动态数组
  - Operators\* :
    - 矩阵与向量之间的运算
    - 重载操作符



# Test

---

- Input file “input.txt”
- Output file “output.txt”



# My FTP

---

- <ftp://10.13.61.167:21>
- Username: DS05
- Password: DS05
  
- Hand in your homework
  - Upload by FTP
  - Send via E-mail



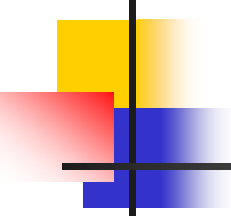
# Resources

- <http://www.math.zju.edu.cn/ligangliu/download/>
- 一些好的编程网站
  - Codeguru: <http://www.codeguru.com/>
  - Codeproject: <http://www.codeproject.com/>
  - VC知识库 : <http://www.vckbase.com/>
  - MSDN中文网站 :  
<http://www.microsoft.com/china/msdn/default.aspx>



# Discussions

---

A decorative graphic in the top-left corner consisting of overlapping yellow, red, and blue squares with a black crosshair.

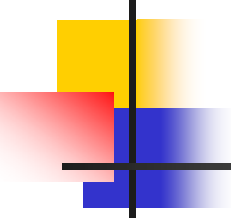
---

```
#include <stdio.h>
void main()
{
    int a=2,b=-1,c=2;
    if(a<b)
    if(b<0) c=0;
    else c+=1;
    printf("%d\n", c);
}
```





[back](#)



```
#include <stdio.h>
void main()
{
    int a=2,b=-1,c=2;
    if(a<b)
    if(b<0) c=0;
    else c+=1;
    printf("%d\n", c);
}
```

```
#include <stdio.h>
void main()
{
    int a=2,b=-1,c=2;
    if(a<b)
    {
        if(b<0)
        {
            c=0;
        }
        else
            c+=1;
    }
}
```

```
#include <stdio.h>
void main()
{
    int a=2,b=-1,c=2;
    if(a<b)
    {
        if(b<0) c=0;
    }
    else c+=1;
    printf("%d\n", c);
}
```