

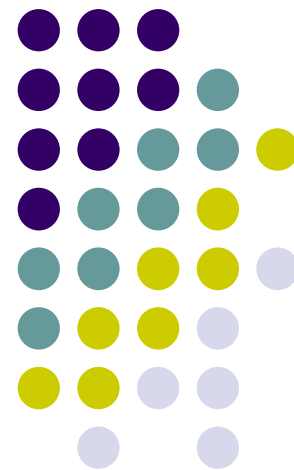
程序设计基础知识难点 和教学方法探讨

南京航空航天大学

计算机学院

皮德常

2015. 12. 05





主要内容

一、知识难点与教学方法探讨

二、课外培养学生的创新能力

三、科研与教学关系的思考



一、知识难点与教学方法探讨

大学培养的人才应是具有创新能力的复合型人才。由于科学技术的发展，计算类学科与其它学科间交叉、融合日益增强。

在多数大学的培养目标中，学生的程序设计能力应是一个重要因素。



例2: 【例2.13】 求100~150之间的所有素数。

1. 思考算法中部分知识点的来源

分析循环 2 到 \sqrt{x} 的原因:

设 x 不是一个素数, 那么 x 就应当有一个不小于 2 的因子 m , 即 x 可以分解为: $x=(\sqrt{x})^2 = m.n$, 其中 m 和 n 为 x 的两个约数, 显然 m 和 n 都是大于或等于 2 的整数。假设 m 是两个数中比较小的一个整数, 则可以推出: $(\sqrt{x})^2 = m.n \geq m^2 \geq 2^2$, 由前提条件“ x 是一个正整数”可得: $\sqrt{x} \geq m \geq 2$ 。故我们可以得到一个结论: 若正整数 x 不是一个素数, 那么在 $2 \sim \sqrt{x}$ 之间必有一个约数。



例：递归调用

【例】利用递归求n!

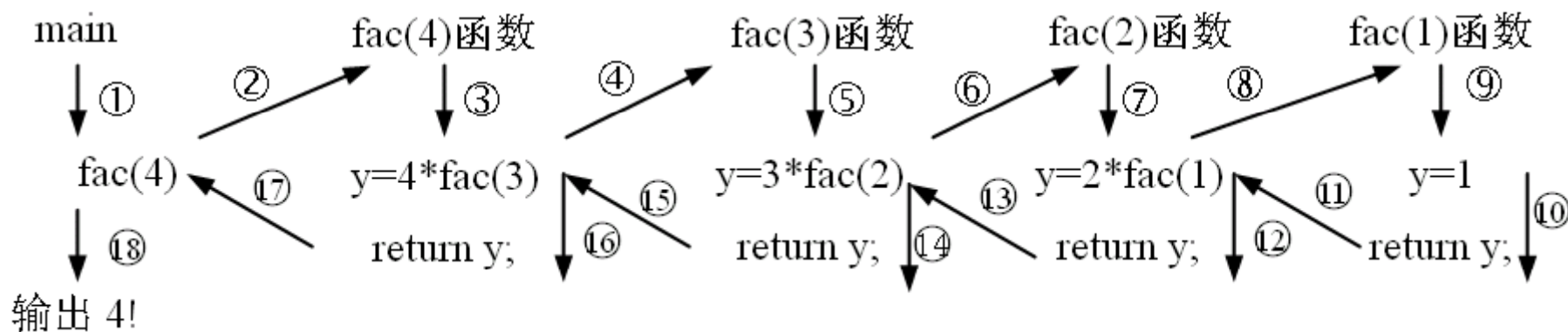


图 3.5 求 n! 的递归调用图



递归调用过程：栈

n=4
result=?

a. 执行第 10 行

y=?
n=4
返回地址 1
n=4
result=?

b. 执行第 22 行

y=?
n=3
返回地址 2
y=?
n=4
返回地址 1
n=4
result=?

c. 执行第 22 行

y=?
n=2
返回地址 3
y=?
n=3
返回地址 2
y=?
n=4
返回地址 1
n=4
result=?

d. 执行第 22 行

y=1
n=1
返回地址 4
y=?
n=2
返回地址 3
y=?
n=3
返回地址 2
y=?
n=4
返回地址 1
n=4
result=?

e. 执行第 20 行



```
void cat(int x)
{
    cout<< x << endl;
}
void dog(int a)
{
    int b=10;
    cat(b+a);
}
int main ( )
{
    int a=20;
    dog(a);
    cout << a << endl;
    return 0;
}
```

例：局部变量与栈

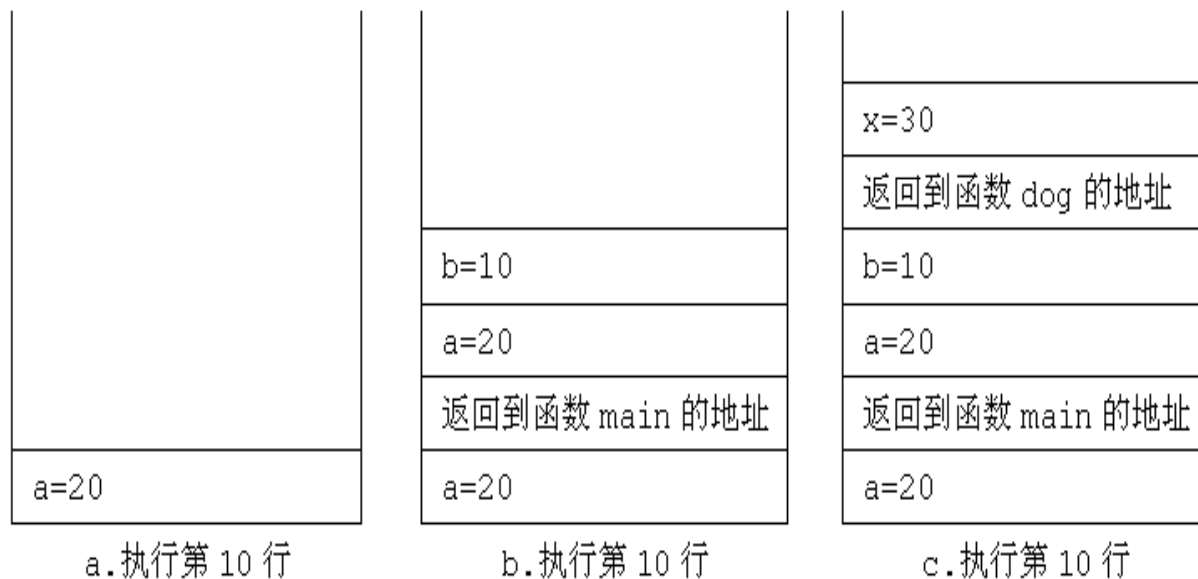


图 3.3 函数调用与内存分配

结论：通过栈机制，任何一个函数均可以定义自己的局部变量，而不用关心其它函数是否存在同名变量。



一、知识难点与教学方法探讨

2. 横向联想

【例】递归：汉诺塔

(1) 若将 n 个盘子从A柱子移到C柱子，要移动 2^n-1 次。

采用**数学归纳法**证明。

(2) 能证明模糊中的“光头问题”吗？

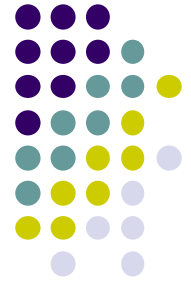


一、知识难点与教学方法探讨

观点：兼顾程序设计语言的实现原理，注重培养学生举一反三和思考问题的能力，乃至提出问题的能力。

◆学生最缺乏的**创新意识**；

◆教师在授课中缺乏对学生创新、好奇的引导，提高创新意识。



一、知识难点与教学方法探讨

爱因斯坦：“提出一个问题往往比解决一个问题更重要，因为解决一个问题也许是一个数学上或实验上的技能而已。而提出新的问题，新的可能性，从新的角度去看旧的问题，却需要有创造性的想象力。而且标志着科学的真正进步。”



一、知识难点与教学方法探讨

3. 纵向分析

Java与C++中参数传递的区别

<pre>package chapter2; class test{ int x; } public class Test_Paras { static test fun(test o,int y){ o.x=20; y=1; return o; } }</pre>	<pre>public static void main(String [] args){ test obj1 , obj2; int Val = 3; obj1=new test() ; obj1.x=100; obj2=fun(obj1,Val); System.out.println("obj1.x="+obj1.x); System.out.println("obj2.x="+obj2.x); System.out.println("Val="+Val); }</pre>
--	---



一、知识难点与教学方法探讨的探讨

3. 纵向分析

指针变量做函数形参

```
int main ( )  
{  
    int x=10, y=20, *p1, *p2;  
    p1=&x;  
    p2=&y;  
    cout << x << "\t" << y << endl;  
    swap( p1, p2 );  
    cout << x << "\t" << y << endl;  
  
    return 0;  
}
```

```
void swap( int *px, int *py )  
{  
    int t;  
  
    t=*px ;  
    *px=*py ;  
    *py= t;  
}
```



一、知识难点与教学方法探讨的探讨

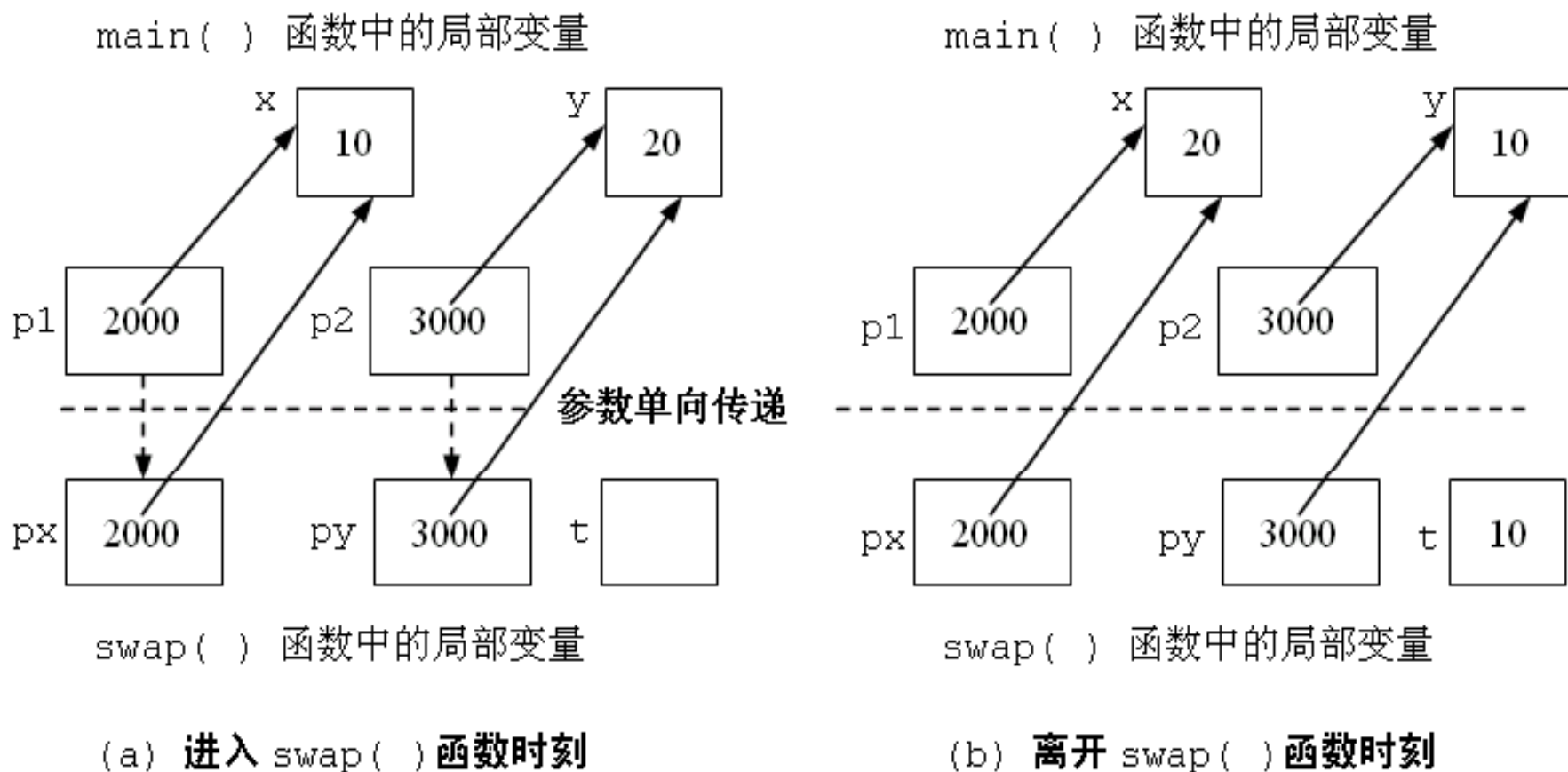


图 5-12 指针变量做参数



一、知识难点与教学方法探讨的探讨

结论： C++中参数传递**只有两种**，按值传递和按引用传递，没有按指针传递。



一、知识难点与教学方法探讨

4. 类比：设法调动学生的积极性，尤其是思考的积极性。

**【例】二分查找算法
与
“非常6+1”**



一、知识难点与教学方法探讨

强调程序设计风格：

例：第一个程序，就明确要求学生要遵循常规。



- 类名、函数名和变量名的定义做到“见名知义”；

- 采用缩排格式组织程序代码和尽可能多地注释。

- 对编程题的要求：每个编程题，都要在每个习题的开头部分都要加上注释。注释包含：你的姓名、编程日期、习题所属章节、题号和题目名称。

```
//*****  
//* 作者：Your Name  
//* 日期：20YY年MM月DD日  
//* 章节：第X章  
//* 题号：习题N  
//* 题目：在屏幕上显示 “Programming is fun!”  
//*****
```



例1：反对晦涩难懂的语句

```
int num=0;
```

```
while( num++ < 10)
```

```
    cout << num << endl;
```

```
cout << num << endl;
```



例2：强调实用

```
int *pInt ;  
pInt = new int [100] ;  
if ( pInt = NULL )  
    . . .
```

```
int *pInt ;  
pInt = new int [100] ;  
if ( NULL = pInt )  
    . . .
```

```
int score=59;  
if ( score = 99 )  
    cout<<"高分啊！";
```

```
int score=59;  
if ( 99 = score )  
    cout<<"高分啊！";
```



一、知识难点与教学方法探讨

思考、再思考

思考1: 输出二维数组的地址。

```
int main ( )
{   int a[3][4]={ {1, 2, 3, 4 },
                  {5, 6, 7, 8 },
                  {9,10,11,12 }
                };
    cout<< a[0] <<"\t"<< a <<"\t"<< &a <<endl;
    cout<<a+1<<"\t"<<a[1]<<"\t"<<*(a+1)<<"\t"
        <<&a[1]<<endl;
    cout<<a[1]+2<<"\t"<<*(a+1)+2<<"\t"
        <<&a[1][2]<<"\t"<<endl;
    return 0;
}
```



一、知识难点与教学方法探讨

思考2：基于对调用拷贝构造函数的认识，给出关于编写优质程序的建议：如果函数返回值是一个对象，要考虑return语句的效率。以String对象为例：

```
return string ( s1 + s2 );
```

与如下方法相比：

```
string temp ( s1 + s2 );
```

```
return temp;
```



一、知识难点与教学方法探讨

5. “思考→总结→创新”

```
class Base{
    int x=1;
    void print( ){
        System.out.println("当前类为 "+ this.getClass( ).getName( ));
        System.out.println("对象的x= "+this. x );
    }
}

class Derived extends Base{
    int x=2;
    void print( ){
        System.out.println("当前类为 "+ this.getClass( ).getName( ));
        System.out.println("对象的x= "+this. x );
    }
}
```

```

public class confusions{
    public static void main(String [ ] args){
        Base obj = new Derived( );

        obj.print( ); // 哪一个print ?
        System.out.println("对象的x= "+ obj.x ); // 哪一个x?
    }
}

```

子类本身定义

```

int x = 2
void print( )

```

从父类继承部分

```

int x = 1
void print( )

```

总结

1. Java的多态分为编译多态和运行多态两种:

(1) 编译多态: 方法重载、方法覆盖

```
Derived o1 = new Derived( );
```

```
o1.print( );
```

(2) 运行多态:

```
Base o2 = new Derived( );
```

```
o2.print( );
```

2. 多态性仅适用于方法, 而不适合于实例变量;



一、知识难点与教学方法探讨的探讨

6. “综合即创造”

main 函数的参数

【例1】 通过命令行参数计算输入数据的和。



一、知识难点与教学方法探讨的探讨

```
int main ( int argc, char *argv[ ] )
{
    int sum, i;

    cout << "Command name:" << argv[0] << endl;
    for(sum=0, i=1; i<argc; i++)
        sum += atoi(argv[i]);
    cout << "Sum is:" << sum << endl;

    return 0;
}
```



一、知识难点与教学方法探讨的探讨

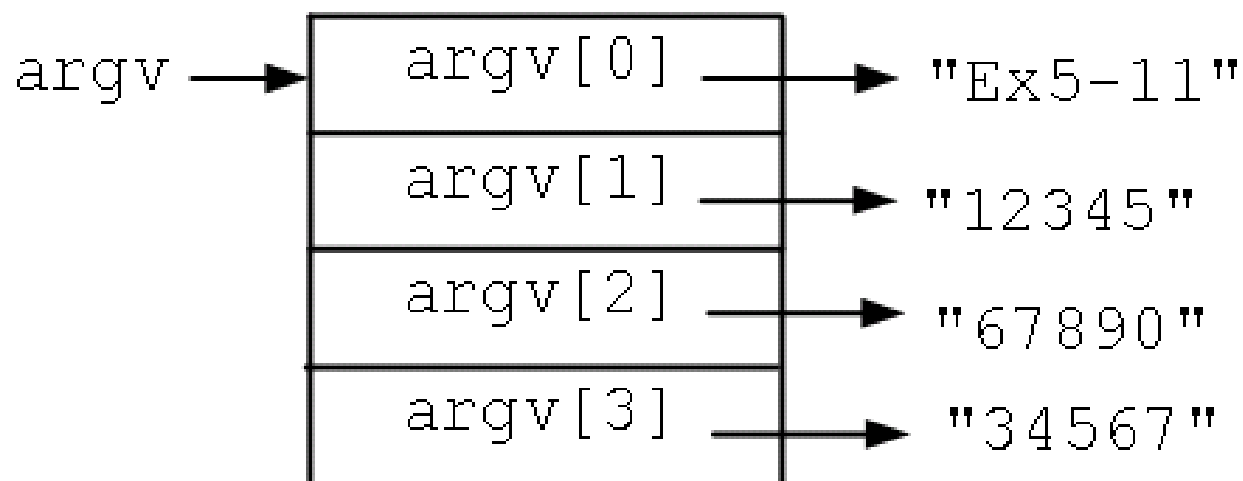


图 5-16 main 函数参数的含义

思考：针对上图，思考如下程序段的输出结果

```
cout << ++ * ++ * ++ argv << endl;  
cout << ++ * ++ * argv << endl;
```



一、知识难点与教学方法探讨

7. 易混淆的概念的总结与类比

1. `int *p;`
2. `int (*p)[10];`
3. `int * p[10];`
4. `int **p;`
5. `int (*p)(int, int);`
6. `int *fun(int, int);`
7. NULL指针
8. 野指针

类比是一种推理模式，是根据两个不同对象在某方面的相似，推测出这两个对象在其它方面也可能存在相似之处。

类比推理可启迪思维、促进联想、拓展认识。

波利亚（George Polya, 1887-1985）

：“类比是一个伟大的引路人”。



一、知识难点与教学方法探讨的探讨

8. 故意找茬，引导学生思考

```
#include <iostream>
#include <fstream>
#include <cstdlib>
using namespace std;
int main( )
{
    char ch, again;
    long pos;
    fstream file("digit.txt", ios::in);
    if( file.fail( ))
    {
        cout << "打开文件digit.txt出错! \n" ;
        exit( 0 );
    }
}
```

设内容为: abcdefg1234567890



一、知识难点与教学方法探讨的探讨

```
do
{
    cout << "输入一个要读字符的位置： " ;
    cin >> pos;
    file.seekg(pos, ios::beg);
    file.get(ch);
    cout << pos << "号位置上的字符是： " << ch << endl;

    cout << "\n还继续读取吗(Y/N) ? ";
    cin >> again;
}while( toupper(again) == 'Y' );
file.close( );
return 0;
}
```

设内容为： abcdefg1234567890



一、知识难点与教学方法探讨的探讨

运行结果：

输入一个要读字符的位置： 8

8 号位置上的字符是： 2

设内容为： abcdefg1234567890

还继续读取吗(Y/N) ? y

输入一个要读字符的位置： 1000

1000 号位置上的字符是： 2

还继续读取吗(Y/N) ? y

输入一个要读字符的位置： 9

9 号位置上的字符是： 2

从此开始，一错到底，why?

还继续读取吗(Y/N) ? n



一、知识难点与教学方法探讨的探讨

本门课的课程设计：

- 课程设计对这门课的重要性；
- 代码量。0.3~1.2万行

教指委一般要求：**4万=0.5+1+1+1+1.5万**

最低要求：**2万=0.2+0.4+0.6+0.8万**



一、知识难点与教学方法探讨

课堂教学法总结：“句号+问号”

对于原理性知识直接讲解，而在程序设计中采用提出问题，引导学生分析问题，然后得出结论的句号教学法。



一、知识难点与教学方法探讨

华罗庚的“兔子”理论：

本科生：学习捡“死”兔子。

硕士生：学习打一只在视野中奔跑的活兔子。

博士生：学习打一只看不到的活兔子。



主要内容

一、知识难点与教学方法探讨

二、课外培养学生的创新能力

三、科研与教学关系的思考



二、课外培养学生创新能力

1. 大学生科创项目驱动：

●学校：科技创新基金每年20万元；

●学院：20万元。

覆盖面60%。鼓励学生进行自主学习、自主创新和交叉研究。

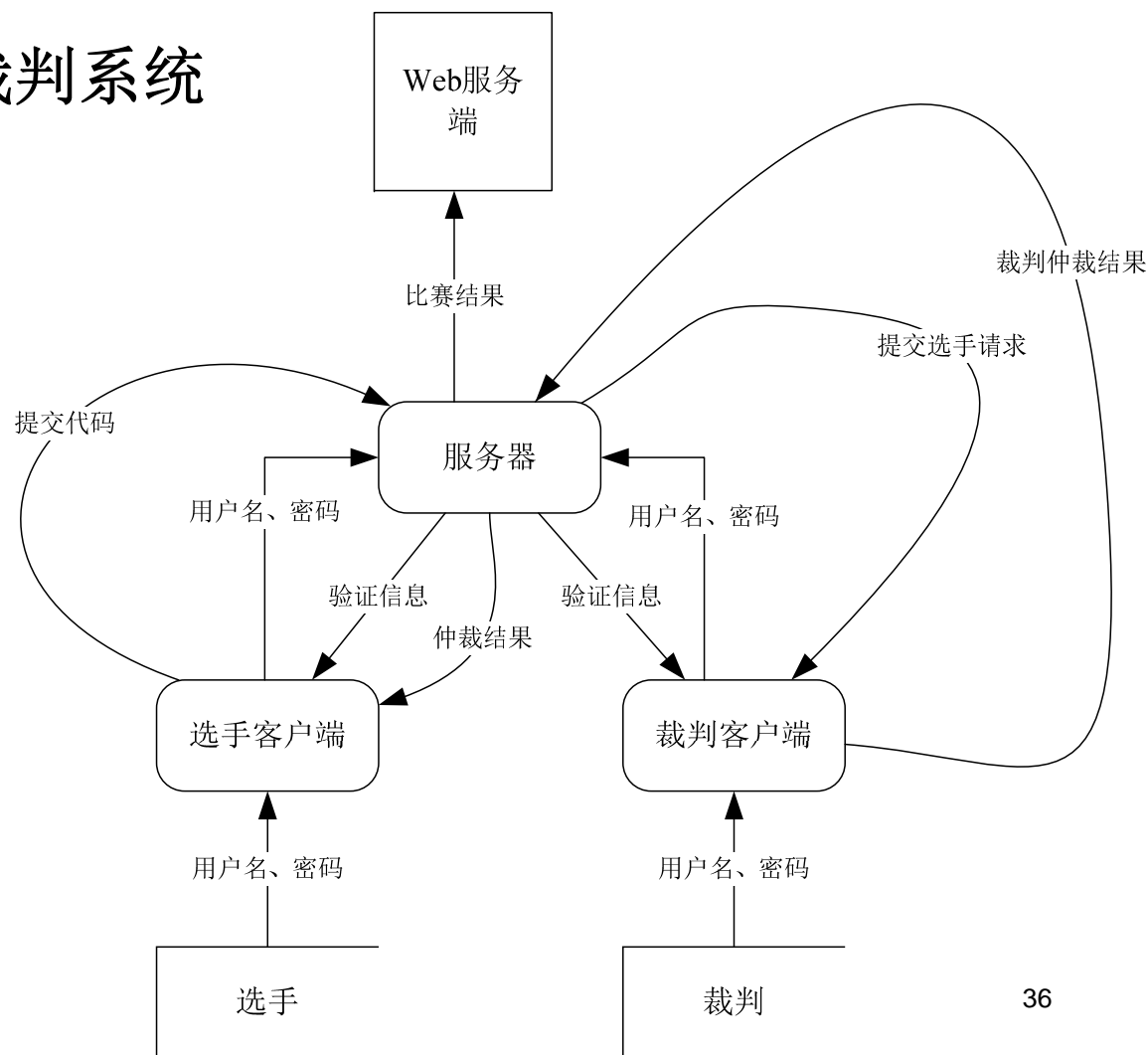
鼓励学生抓着机会，利用基金，进行科创。



二、课外培养学生创新能力

(1) 题目：ACM比赛裁判系统

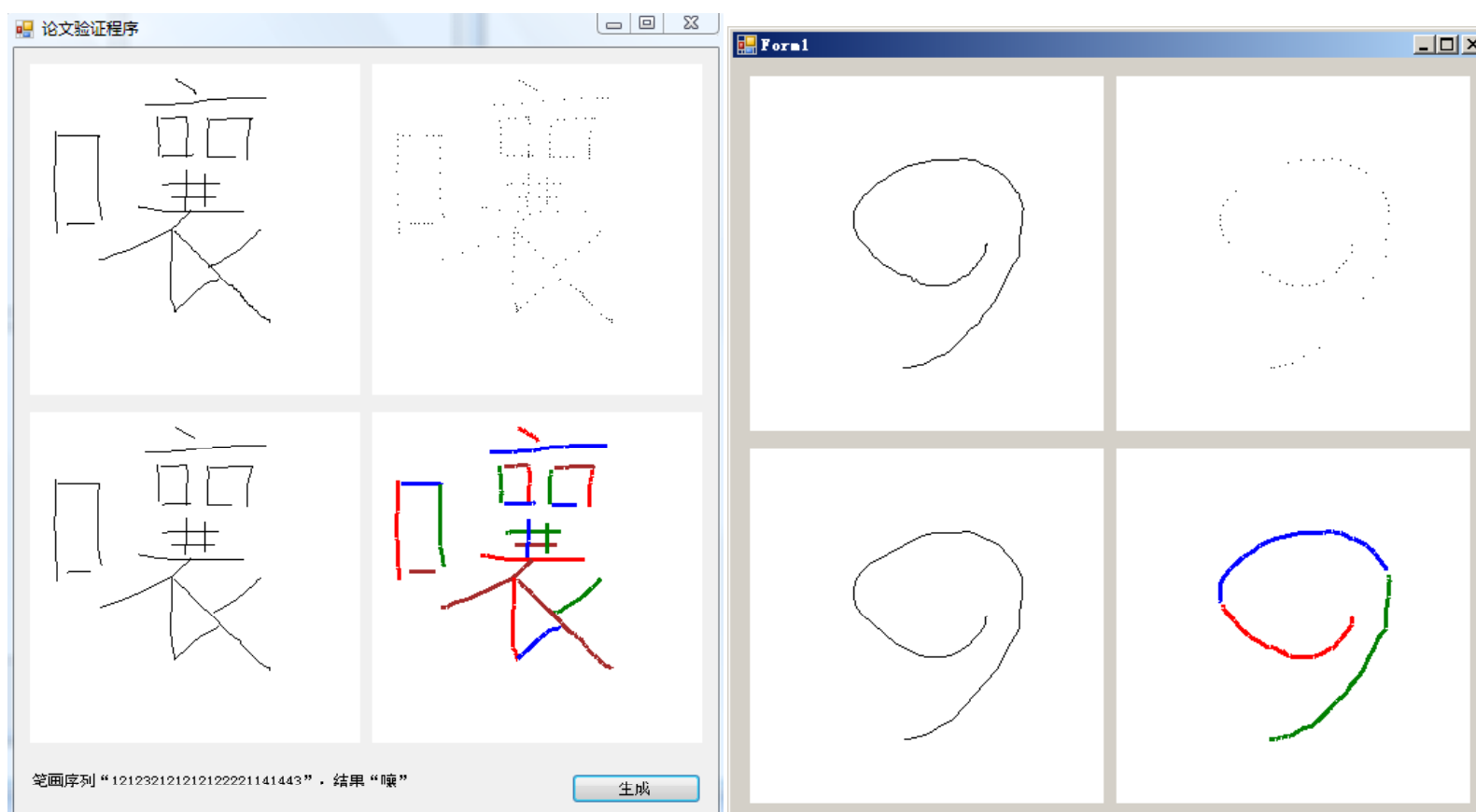
系统由：服务器端，选手端和裁判端组成。





二、课外培养学生创新能力

(2) 基于拟合的二值图像矢量化识别算法

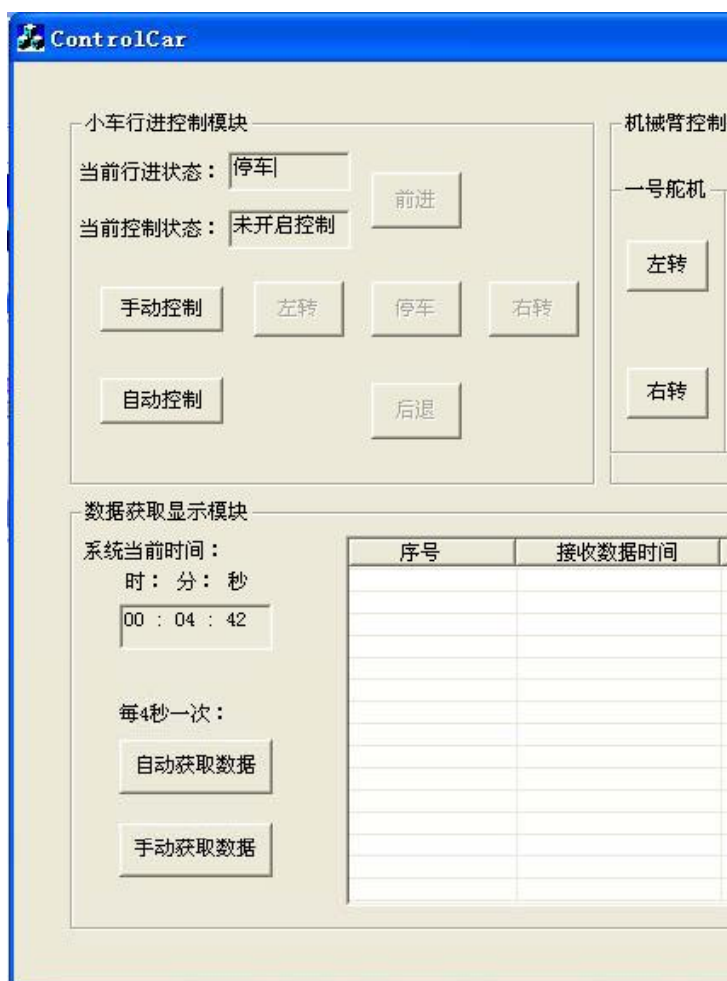


识别手写汉字“嚷”和五线谱低音符



二、课外培养学生创新能力

(3) 远程控制机器人。





二、课外培养学生创新能力

(4) “D程序设计语言编译器的设计与实现”的失败

D语言简介：

D语言是由 Digital Mars公司的Walter Bright所开发程序设计语言。D语言起源于C++，属静态类型的语言，语法与Java或者C#非常类似。它不仅具有C++的一些功能，而且引入其他其它语言，如Java，C#，Python和Ruby的某些功能。



二、课外培养学生创新能力

不可否认，世上无难事，只怕有心人。可是也必须承认，有些事情，仅仅是有心是不够的，还要有这个能力。

我就是对自己的预估过高，以一种年轻气盛的方式看待问题，然后轻易的就决定实现这个项目，结果将自己和组员推向一个尴尬的境地。

通过一些调查发现，写一个编译器，那是需要具有大能耐、大智慧，才能够完成的。像我这种毛头小孩，暂时没有这种能耐。当然，无可否认，这也让我学到了很多东西，让我以后能够以一种更实际的眼光看待问题。我的组员也是学到一些东西，比如不要輕易被我这种人拉下水。

当然，不可否认，也是做过一些努力来完成这个项目。可是，那却如撼山一般。。。



二、课外培养学生创新能力

2. 加强舆论、宣传和引导。

学生原话：“兴趣是最好的老师，。 。 。 。 。”



主要内容

一、知识难点与教学方法探讨

二、课外培养学生的创新能力

三、科研与教学关系的思考



三、科研与教学关系的思考

- 将科研成果及时转化为教学，把最新研究成果和方法，在第一时间传授给学生，可使学生及时了解某个领域的理论与方法，丰富课堂教学。
- 科研项目可改善实验条件，为人才培养提供了有利的硬件环境，可建成实验室，为人才培养提供实验环境。
- 科研可提高的学生实践能力，将科研课题的研究成果融进本科生培养过程，可参加学科竞赛，甚至学生获奖。



三、科研与教学关系的思考

观点：

- 教学是我们的生存之本，科研是“强教”之路。
- 以“传授知识、培养能力、提高素质、培养做人”为已任。
- 采用“类比、启发、科创”等手段，提高学生的学习兴趣，达到教与学的目的。



三、科研与教学关系的思考

观点：

- 我们所从事教师职业：不仅是我们谋生的手段，更重要的是，我们在从事一个事业——培育人（我们的工作事业）。人是世界上最高级的动物，而我们就是塑造和培养人，还有哪个职业能比此行更高尚、更伟大呢？



谢谢大家!