C/C++程序设计案例实战

——学生管理系统之班级最高分

华中农业大学信息学院　章 英

问题引入

如何录入数据

学号	分数
2019010	85
2019002	90
2019032	76
2019005	88
2019028	92
…….	…….
2019006	89

如何存储数据

问题分析

如何录入数据

学号	分数
2019010	85
2019002	90
2019032	76
2019005	88
2019028	92
…….	…….
2019006	89

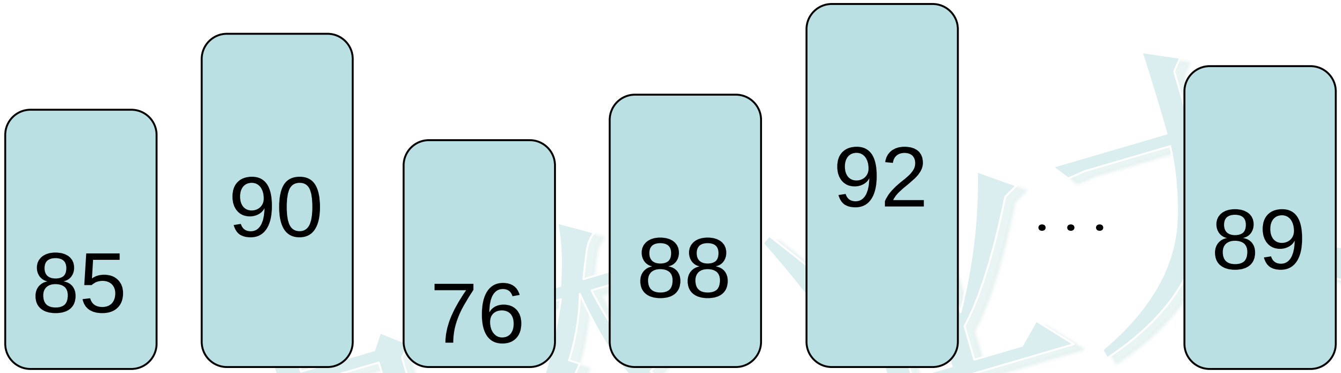
如何存储数据

```
long num[35];   int score[35]
```

数据录入

```
#include<iostream>
using namespace std;
#define N 35
int main()
{
  long num[N],maxNum;
  int score[N],maxScore;
  int n,i;
  cout<<"How many students?";
  cin>>n;
  cout<<"Input ID and score:\n";
  for (i=0; i<n; i++)
  {
    cin>>num[i]>>score[i];
  }
... ...
```

数据查找



85 90 76 88 92 . . . 89

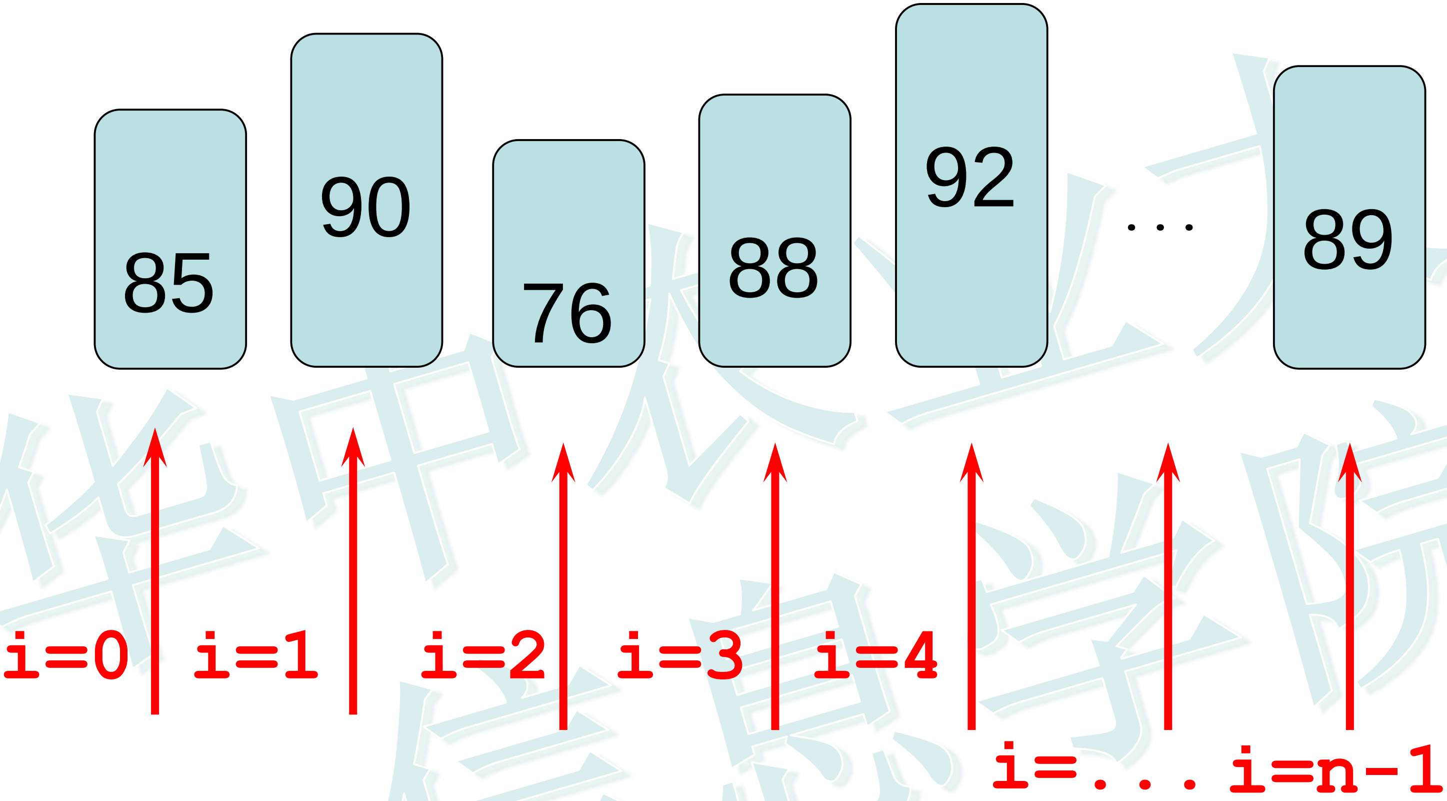
二分查找

B树查找

哈希查找

……

数据查找



85

90

76

88

92

…

89

i=0 i=1 i=2 i=3 i=4

i=… i=n-1

pMaxScore 92

pMaxNum 2019028

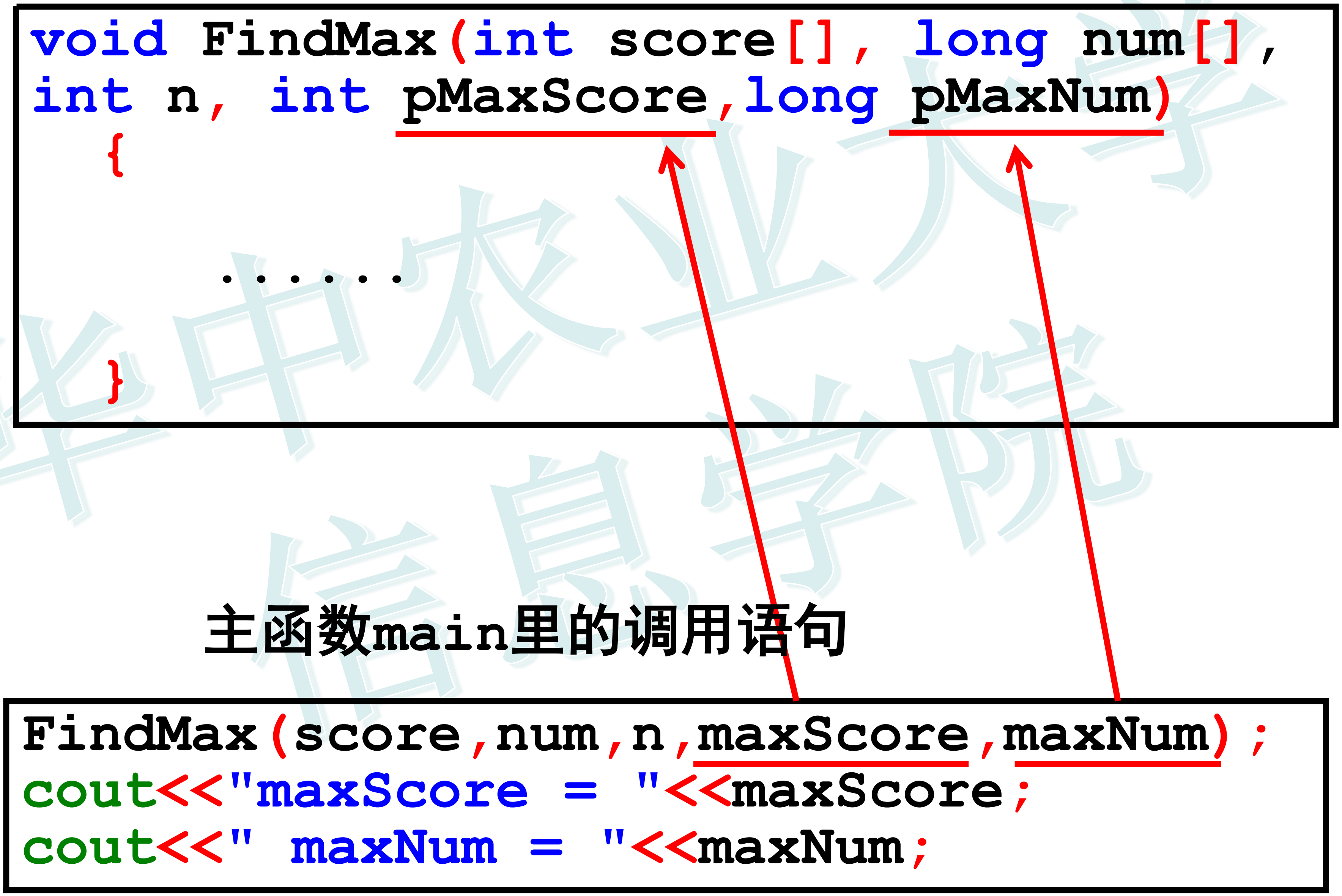
数据查找——代码实现

```
pMaxScore = score[0];
pMaxNum = num[0];
for (i=1; i<n; i++)
{
  if (score[i] > pMaxScore)
  {
      pMaxScore = score[i];
      pMaxNum = num[i];
  }
}
```

查找函数——功能模块化

```
void FindMax(int score[],long num[],
int n, int pMaxScore,long pMaxNum)
{ int i;
  pMaxScore = score[0];
  pMaxNum = num[0];
  for (i=1; i<n; i++)
  {
    if (score[i] > pMaxScore)
    {
        pMaxScore = score[i];
        pMaxNum = num[i];
    }
  }
}
```

查找函数——功能模块化

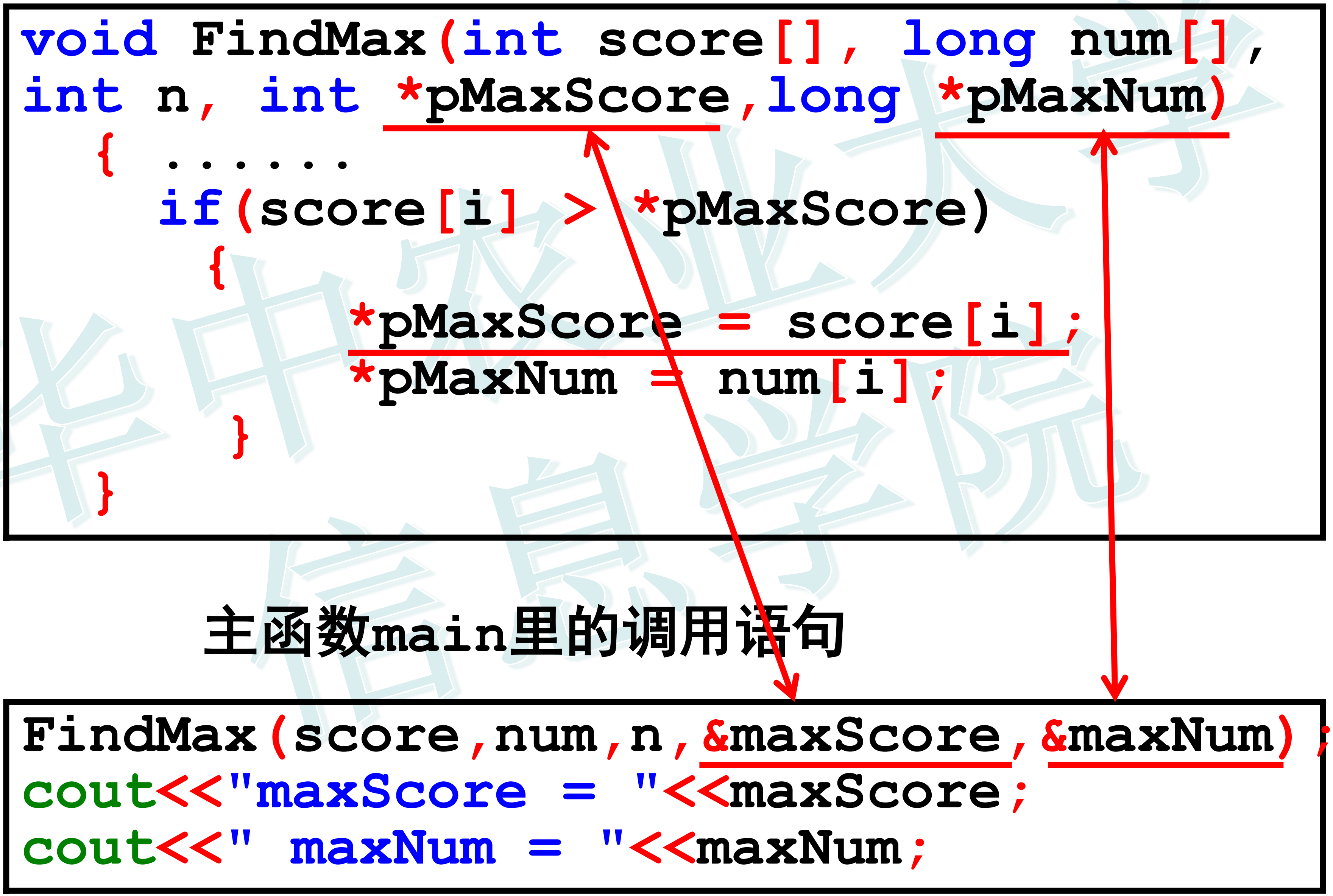


```
void FindMax(int score[], long num[],
int n, int pMaxScore,long pMaxNum)
   {

      ......

   }
```

主函数main里的调用语句

```
FindMax(score,num,n,maxScore,maxNum);
cout<<"maxScore = "<<maxScore;
cout<<" maxNum = "<<maxNum;
```

修改后



```
void FindMax(int score[], long num[],
int n, int *pMaxScore,long *pMaxNum)
   { ......
     if(score[i] > *pMaxScore)
       {
           *pMaxScore = score[i];
           *pMaxNum = num[i];
        }
   }
```

主函数main里的调用语句

```
FindMax(score,num,n,&maxScore,&maxNum);
cout<<"maxScore = "<<maxScore;
cout<<" maxNum = "<<maxNum;
```

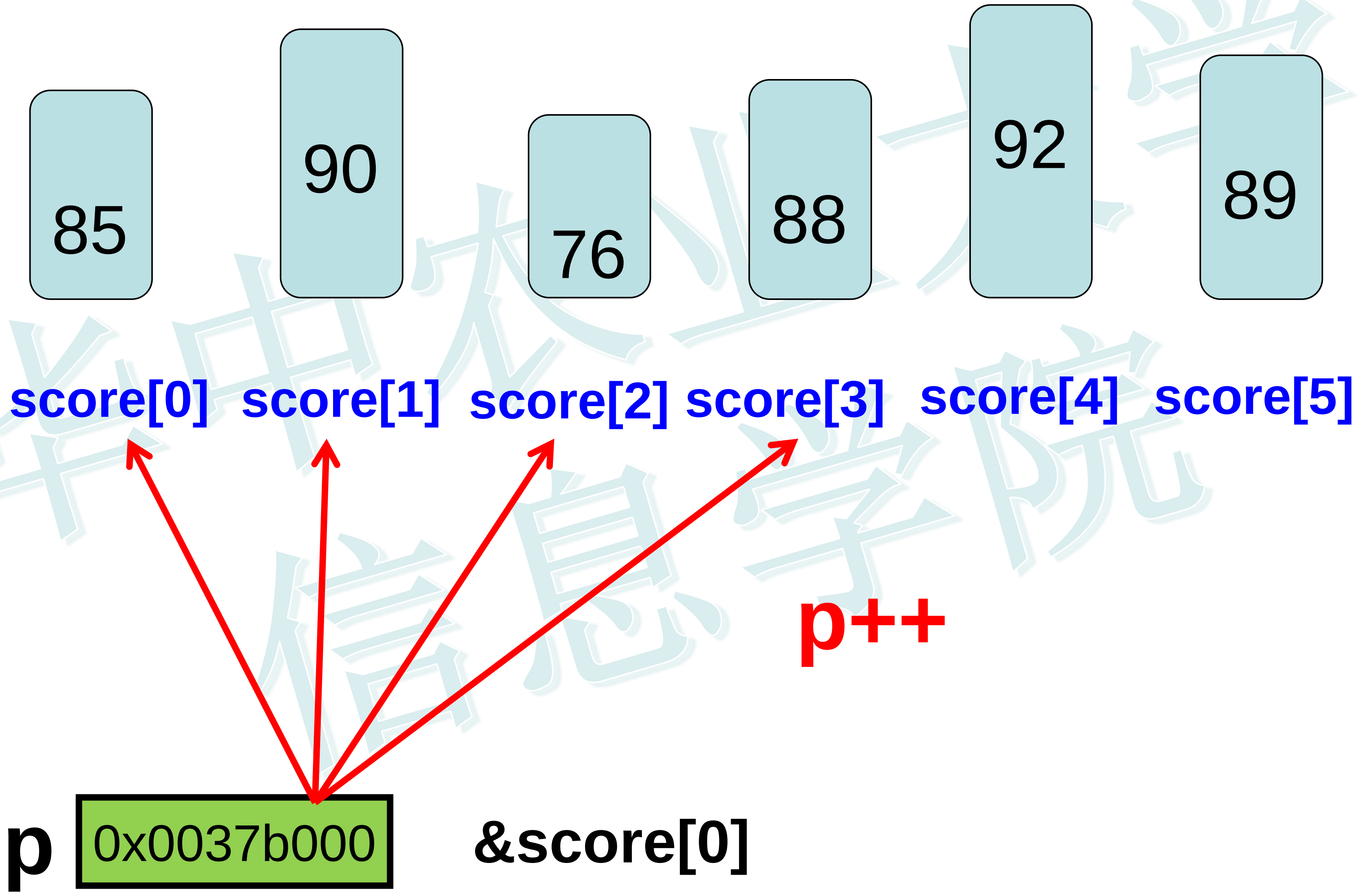
变量对比

主函数main	子函数FindMax
score[ ]	score[ ]
num[ ]	num[ ]
n	n
maxScore	*pMaxScore
maxNum	*pMaxNum

案例进阶

主函数main	子函数FindMax
score[ ]	*p
num[ ]	*q
n	n
maxScore	*pMaxScore
maxNum	*pMaxNum

指针与一维数组



85 90 76 88 92 89

score[0] score[1] score[2] score[3] score[4] score[5]

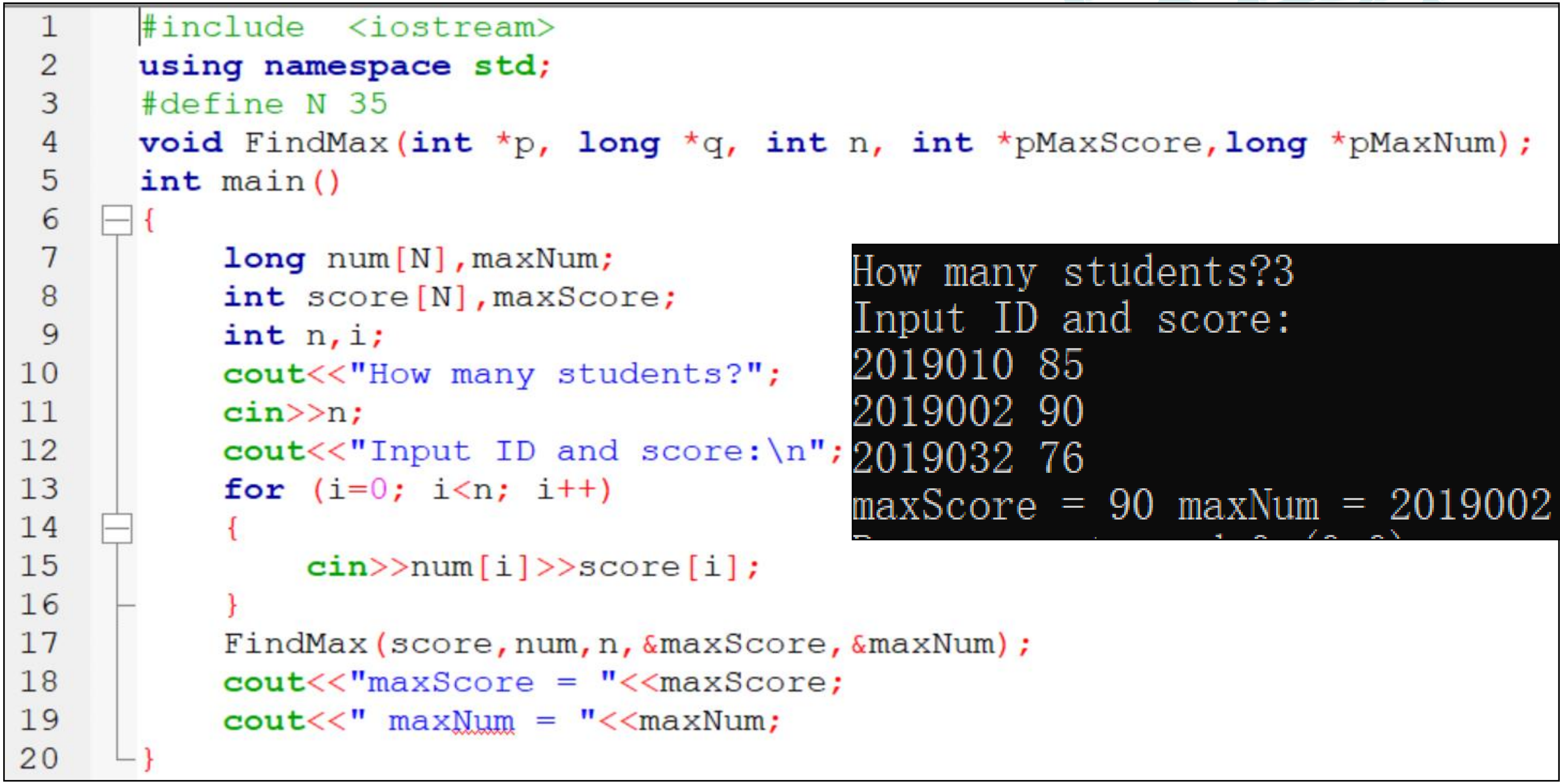
p++

p 0x0037b000 &score[0]

案例进阶——代码实现

```
void FindMax(int *p, long *q,
int n, int *pMaxScore,long *pMaxNum)
{ int i;
  *pMaxScore= *p;//*(&score[0])、p[0]
  *pMaxNum = *q;
  for (i=1; i<n; i++)
  {
    p++; q++;
    if(*p>*pMaxScore)//p[i]、score[i]
     {
       *pMaxScore = *p;
       *pMaxNum = *q;   //num[i]、q[i]
     }
  }
}
```

案例进阶——代码实现



```cpp
#include  <iostream>
using namespace std;
#define N 35
void FindMax(int *p, long *q, int n, int *pMaxScore,long *pMaxNum);
int main()
{
    long num[N],maxNum;
    int score[N],maxScore;
    int n,i;
    cout<<"How many students?";
    cin>>n;
    cout<<"Input ID and score:\n";
    for (i=0; i<n; i++)
    {
        cin>>num[i]>>score[i];
    }
    FindMax(score,num,n,&maxScore,&maxNum);
    cout<<"maxScore = "<<maxScore;
    cout<<" maxNum = "<<maxNum;
}
```

```
How many students?3
Input ID and score:
2019010 85
2019002 90
2019032 76
maxScore = 90 maxNum = 2019002
```

等价关系

	数组值	地址
第一个 数组元素	score[0]	score
	*p	p
	*(p+0)	&score[0]
第i个 数组元素	score[i]	&score[i]
	*(&score[i])	&score[i]
	*(score+i)	score+i
	*(p+i)	p+i

小结

（1）能够编写形参为指针的函数

（2）能够分析形参为指针、实参为一维数组的函数中指针与数据之间的指向关系

延伸

请修改本案例代码，将数据录入进行模块化设计，使用指针做参数，实现数据的录入。