



西北工业大学

NORTHWESTERN POLYTECHNICAL UNIVERSITY

C++程序设计

Programming in C++



1011018

主讲：魏英，计算机学院

指针与数组

2、指针与字符串

- ▶ 可以利用一个字符型的指针处理字符串，其过程与通过指针访问数组元素相同。使用指针可以简化字符串的处理，是程序员处理字符串常用的编程方法。

- ▶ C++ 允许定义一个字符指针，初始化时指向一个字符串常量，一般形式为：

```
char *p="C Language";
```

```
char *p;  
p="C Language";
```

- ▶ 初始化时，p存储了这个字符串首字符地址4000，而不是字符串常量本身，称p指向字符串。

- ▶ 通过字符指针可以访问字符串。例如：

```
char str[]="C Language", *p=str; //p指向字符串的指针
cout<<p<<endl; //输出: C Language
cout<<p+2<<endl; //输出: Language
cout<<&str[7]<<endl; //输出: age
```

- ▶ 通过字符指针遍历字符串。

```
char str[]="C Language", *p=str; //p指向字符串的指针
while (*p!='\0') cout<<*p++;
```

18.2 指针与字符串

【例18.2】指针访问字符串举例

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      char str[100], *p=str;
6      cin>>str; //输入字符串
7      while (*p) p++; //指针p指向到字符串结束符
8      cout<<"strlen="<<p-str<<endl;
9      return 0;
10 }
```

运行结果： JavaScript ✓
strlen=10

- ▶ 请记住，指针可以指向数组，使得数组的访问多了一种途径，但指针并不能替代数组来存储大批量数据。

```
char s[100]="Computer";  
char *p="Computer";
```

- ▶ ①存储内容不同
- ▶ ②运算方式不同
- ▶ ③赋值操作不同

- ▶ 如果字符串数组是一个二维字符数组，例如：

```
char sa[6][7]={"C++","Java","C","PHP",  
              "CSharp","Basic"};
```

- ▶ 则字符串数组的内存形式为：

	sa						
sa[0]	C	+	+	\0			
sa[1]	J	a	v	a	\0		
sa[2]	C	\0					
sa[3]	P	H	P	\0			
sa[4]	C	S	h	a	r	p	\0
sa[5]	B	a	s	i	c	\0	

CP 程序设计