

## C 语言函数指针（指向函数的指针）

一个函数总是占用一段连续的内存区域，函数名在表达式中有时也会被转换为该函数所在内存区域的首地址，这和数组名非常类似。我们可以把函数的这个首地址（或称入口地址）赋予一个指针变量，使指针变量指向函数所在的内存区域，然后通过指针变量就可以找到并调用该函数。这种指针就是函数指针。

函数指针的定义形式为：

```
returnType (*pointerName)(param list);
```

returnType 为函数返回值类型，pointerName 为指针名称，param list 为函数参数列表。参数列表中可以同时给出参数的类型和名称，也可以只给出参数的类型，省略参数的名称，这一点和函数原型非常类似。

注意()的优先级高于\*，第一个括号不能省略，如果写作 returnType \*pointerName(param list);就成了函数原型，它表明函数的返回值类型为 returnType。

【实例】用指针来实现对函数的调用。

```
1. #include <stdio.h>
2.
3. //返回两个数中较大的一个
4. int max(int a, int b){
5.     return a>b ? a : b;
6. }
7.
8. int main(){
9.     int x, y, maxval;
10.    //定义函数指针
11.    int (*pmax)(int, int) = max; //也可以写作 int (*pmax)(int a, int
    b)
12.    printf("Input two numbers:");
```

```
13.     scanf("%d %d", &x, &y);
14.     maxval = (*pmax)(x, y);
15.     printf("Max value: %d\n", maxval);
16.
17.     return 0;
18. }
```

运行结果：

Input two numbers:10 50✓

Max value: 50

第 14 行代码对函数进行了调用。pmax 是一个函数指针，在前面加 \* 就表示对它指向的函数进行调用。注意()的优先级高于\*，第一个括号不能省略。