



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

函数的定义和使用

- ◆ 3、函数的声明
- ◆ 4、内联函数

- ▶ 当要调用函数时，C++语言规定在调用一个函数之前必须有该函数的声明。
- ▶ C++规定函数的定义既是函数定义，也是函数声明。换言之，只要函数调用是写在函数定义的后面，就自然有了函数声明。
- ▶ 声明的作用是程序向编译器提供函数的接口信息，因而多次提供接口信息是允许的，但不能提供相互矛盾、语义不一致的接口信息。

9.3 函数的声明

► 如:

```
1  #include <iostream> //标准输入输出函数库
2  #include <cmath> //数学函数库
3  using namespace std; //使用标准命名空间
4  double root(double x, double y) //root函数求x-y的平方根
5  { if(x>=y) return sqrt(x-y); //只有在x大于等于y时计算x-y的平方根
6    else return 0; //否则返回0
7  }
8  int main() //主函数
9  { double a, b; //定义两个浮点型变量
10    cin >> a >> b ; //输入两个数
11    cout << root(a,b) << endl; //输出a-b的平方根
12    return 0; //主函数正常结束返回0
13 }
```

▶ 1. 函数原型

▶ 函数原型（function prototype）的作用是提供函数调用所必须的接口信息，使编译器能够检查函数调用中可能存在的问题，有两种形式：

▶ ① 第一种形式：

```
返回类型  函数名( 类型1  形参1, 类型2  形参2, ..... );
```

▶ ② 第二种形式：

```
返回类型  函数名( 类型1  , 类型2  , ..... );
```

9.3 函数的声明

```
1  #include <iostream>
2  using namespace std;
3  int gcd(int m, int n); //gcd函数原型, gcd函数声明在前
4  int main()
5  {
6      int m,n;
7      cin>>m>>n;
8      cout<<gcd(m,n)<<endl; //调用时已有gcd函数声明
9      return 0;
10 }
```

9.3 函数的声明

```
11 int gcd(int m, int n) //求最大公约数, gcd函数实现在后
12 {
13     int r;
14     while (n!=0) { //欧几里德算法, 原理是:
15         r = m % n ; //r为m/n的余数
16         m = n ; //则gcd(m,n)=gcd(n,r)=...
17         n = r ; //r=0时n即是gcd
18     }
19     return m;
20 }
```

m=9, n=6

① r=3, m=6, n=3

② r=0, m=3, n=0

▶ 2. 函数调用

- ▶ 有了函数声明，就可以调用函数，有参数函数调用的形式为：

函数名(实参列表)

- ▶ 实参可以是常量、变量、表达式和函数调用，各实参之间用逗号（，）分隔。实参的类型、次序、个数应与形参一致。

9.3 函数的声明

- ▶ 在C++语言中，可以用以下几种方式调用函数。

- ▶ (1) 函数表达式。

```
z = max(x,y);
```

- ▶ (2) 函数调用语句。

```
max(x,y);
```

- ▶ (3) 函数实参

```
m=max(max(x,y),z);
```

- ▶ C++ 提供一种提高函数效率的方法，即在编译时将被调函数的代码直接嵌入到主调函数中，取消调用这个环节。这种嵌入到主调函数中的函数称为内联函数（inline function）。

- ▶ 内联函数的声明是在函数定义的类型前加上inline修饰符，定义形式为：

```
inline 返回类型  函数名(形式参数列表)
{
    函数体
}
```

- ▶ 内联函数中不允许用循环语句和switch语句。
- ▶ 内联函数的声明必须出现在内联函数第一次被调用之前。

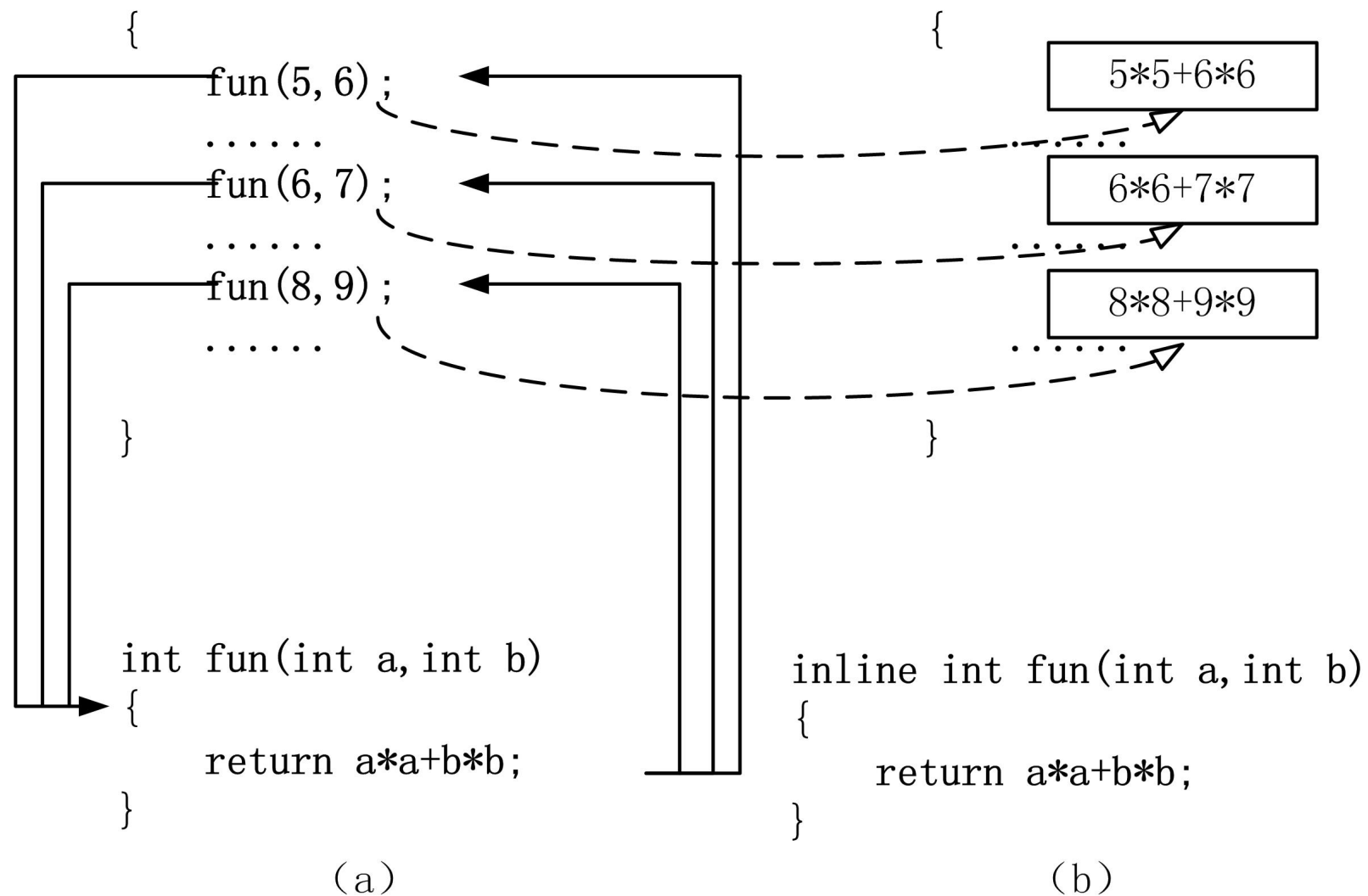
9.4 内联函数

【例9.2】内联函数实例。

```
1  #include <iostream>
2  using namespace std;
3  inline int fun(int a,int b) //内联函数
4  {
5      return a*a+b*b;
6  }
7  int main()
8  {
9      int n=5,m=8,k;
10     k = fun(n,m); //调用点嵌入 a*a+b*b 代码
11     cout<<"k="<<k<<endl;
12     return 0;
13 }
```

9.4 内联函数

图9.1 内联函数调用示意



CP 程序设计