函数

C Reference

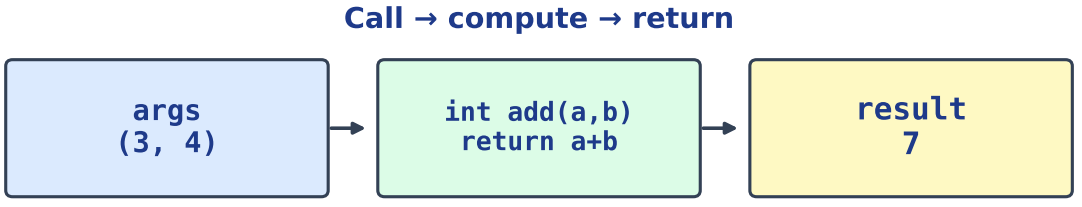
参数与返回值

函数接收参数 (parameter, 即输入), 并交回一个返回值 (return value)。返回类型写在最前面 (`int`、`double`……);`void` 表示不返回任何东西。

```c
#include <stdio.h>

int square(int x) {          // x 是一个参数
    return x * x;            // 交回一个值
}

int main(void) {
    int r = square(5);
    printf("%d\n", r);       // 25
    return 0;
}
```



Exam tip: declare the return type; parameters need types; return ends the function

Parameters in; return value out

函数原型与作用域

C 从上往下读, 所以函数在被调用之前必须先被认识。函数原型 (prototype)— 就是头部那一行加上 ;— 提前声明它, 这样你就能把 `main` 放在最前面。变量的作用域 (scope) 是它所在的代码块: 它是局部 (local) 的, 代码块结束时就消失。

```c
#include <stdio.h>

int add(int a, int b);      // 函数原型: 在使用前声明

int main(void) {
    printf("%d\n", add(3, 4));   // 7
    return 0;
}

int add(int a, int b) {     // 定义放在后面
    int sum = a + b;        // sum 是 add 的局部变量
    return sum;
}
```

常见错误

- 在定义之前使用的函数, 需要在 `main` 上方写原型 (prototype)。
- 参数是**按值**传递 — 函数内的修改不影响调用者, 除非传入指针。
- 在函数内声明的变量是它的局部变量。