

# 程序调试

- 程序错误
  - 语法错误：
  - 语义错误（逻辑错误）：未初始化，`if (i=0)`
- 语法错误编译器检查，人工修改
- 语义错误，代码查错（静态），调试debug（动态）
- 调试之关键：
  - 降速回放程序，察看中间过程，变量变化、流程选择

# 程序调试方法

- 插桩法
  - 添加printf语句
- 调试工具

# 程序调试

- 调试：
  - 重新执行程序
  - 从某行代码慢速执行并观察，类似查看监控
  - 某行代码——断（中断）点
  - 慢速执行——一行一行执行
  - 观察——变量、内存、输出等

# 程序调试

- 查错、排错：
  - 发掘程序中存在的问题
- 步骤：
  - 1. 设置断点
  - 2. 开始调试
  - 3. 单步执行，查看程序运行状态
    - 变量
    - 堆栈等

# 程序调试

- 1. 设置断点
  - 估计程序出错位置，设置断点，让程序运行到断点暂停下来
  - 设置方法：1) 调试菜单命令；2) 鼠标点击；3) 快捷键。
- 2. 启动调试
  - 调试菜单调试命令或快捷键启动，程序运行到断点处暂停，如断点有 `scanf()` 等函数调用，输入数据。
- 3. 单步执行程序
  - 让程序逐条语句执行。方法调试菜单命令或快捷键（`cfree` 中 **F7** 或 **F8**，`VC2010` 中 **F11** 或 **F10**）。
- 4. 观察变量
  - 观察程序中各变量的变化，是否正常，是否与程序逻辑吻合，如是否初始化，类型是否正确等，进而发现问题。
  - 方法：`Cfree` 左下环境窗口，`VC2010` 左下局部变量窗口。

```
#include <stdio.h>
#define FACTOR 0.9f
int main(void){
    int height;
    float weight, stdwt;
    char sex;
    puts("输入身高(cm).");
    scanf("%d", &height);

    puts("输入性别, 男性用m表示, 其它字符表女性.");
    scanf("%c", &sex);

    if (sex == 'm')
        stdwt = (height - 100) * FACTOR;
    else
        stdwt = (height - 100) * FACTOR - 2.5;
    printf("你的标准体重应是%.1fkg\n", stdwt);
    return 0;
}
```