# 笔记

## 第一节

主要内容：**结构体、函数、指针、Keil的使用（包括但不限于J-link、ST-link仿真器的使用方法）。**

1. 结构体：
   1. 意义：增加代码的可读性
   2. 用法：

法1：（培训）

#include <stdio.h>

struct student //结构体类型的说明与定义分开。声明

{

int age; /\*年龄\*/

float score; /\*分数\*/

char sex; /\*性别\*/

};struct student a;//为结构体命名，便于之后引用。

using namespace std;

int main()

{

a.age = 14;

a.score = 96.5;

a.sex = 'f';

printf("年龄：%d 分数：%.2f 性别：%c\n", a.age, a.score, a.sex);

return 0;

}

法2：（网上） 异曲同工

#include <stdio.h>

struct student //结构体类型的说明与定义分开。声明

{

int age; /\*年龄\*/

float score; /\*分数\*/

char sex; /\*性别\*/

};

int main ()

{

struct student a={ 20,79,'f'}; //定义

printf("年龄：%d 分数：%.2f 性别：%c\n", a.age, a.score, a.sex );

return 0;

}

区别在于：培训方法更基于实际，在实际代码中不可能有如此简单直接赋值的情况；而网上方法只给出必要的代码，更注重讲清楚结构体。