<https://blog.csdn.net/stone8761/article/details/50338999>

直接上**示例**了

#include <stdio.h>

struct student\_st

{

char c;

int score;

const char \*name;

};

static void show\_student(struct student\_st \*stu)

{

printf("c = %c, score = %d, name = %s\n", stu->c, stu->score, stu->name);

}

int main(void)

{

// method 1: 按照成员声明的顺序初始化

struct student\_st s1 = {'A', 91, "Alan"};

show\_student(&s1);

// method 2: 指定初始化，成员顺序可以不定，Linux 内核多采用此方式

struct student\_st s2 =

{

.name = "YunYun",

.c = 'B',

.score = 92,

};

show\_student(&s2);

// method 3: 指定初始化，成员顺序可以不定

struct student\_st s3 =

{

c: 'C',

score: 93,

name: "Wood",

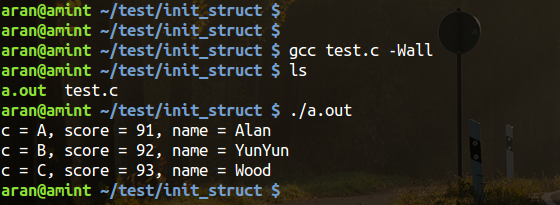
};

show\_student(&s3);

return 0;

}

**运行结果**：



如果想初始化**结构体数组**，可采用 {{ }, { }, { }} 方式，如

struct student\_st stus[2] =

{

{

.c = 'D',

.score = 94,

/\*也可以只初始化部分成员\*/

},

{

.c = 'D',

.score = 94,

.name = "Xxx"

},

};