

FEEL THE MEANING OF THE TRIP

青 / 春 / 不 / 老 / 梦 / 想 / 永 / 在

DREAM  
MY DREAM WILL NEVER STOP

# 高级语言程序设计

## 实验五 编码规范&学生成绩管理系统V1.0

GO!  
TAKE YOU ON A TRIP



哈尔滨工业大学(深圳)  
HARBIN INSTITUTE OF TECHNOLOGY, SHENZHEN

探索 从未停止

# 目录

CONTENT

01

实验回顾

02

编码规范

03

实验目的

04

实验内容

# 实验回顾

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main()
{
    int a, b;

    srand(time(NULL));
    printf("1.time(NULL) is %ld\n", time(NULL));

    a = rand();
    b = rand();
    printf("a=%d, b=%d\n", a, b);

    //////////////////////////////////////

    srand(time(NULL));
    printf("2.time(NULL) is %ld\n", time(NULL));
    a = rand();
    printf("a=%d\n", a);

    srand(time(NULL));
    printf("3.time(NULL) is %ld\n", time(NULL));
    b = rand();
    printf("b=%d\n", b);

    return 0;
}
```

time()函数返回自纪元 Epoch  
(1970-01-01 00:00:00 UTC)  
起经过的时间，以秒为单位

D:\CodeBlocks\WorkSpace\2019\_3C\randomTest\bin\Debug\randomTest.exe

1.time(NULL) is 1602073268  
a=23873, b=18267

2.time(NULL) is 1602073268  
a=23873

3.time(NULL) is 1602073268  
b=23873

Process returned 0 (0x0) execution time : 0.628  
Press any key to continue.





# 实验回顾

---

## ➤ 随机数

随机数其实是伪随机数，之所以是“伪”，因为这些数是有规律的，只不过因为算法复杂，很难看出来。

所谓巧妇难为无米之炊，再厉害的算法，没有一个初始值，它也不可能凭空造出一系列随机数来。所以，我们需要“种子”，也就是初始值。

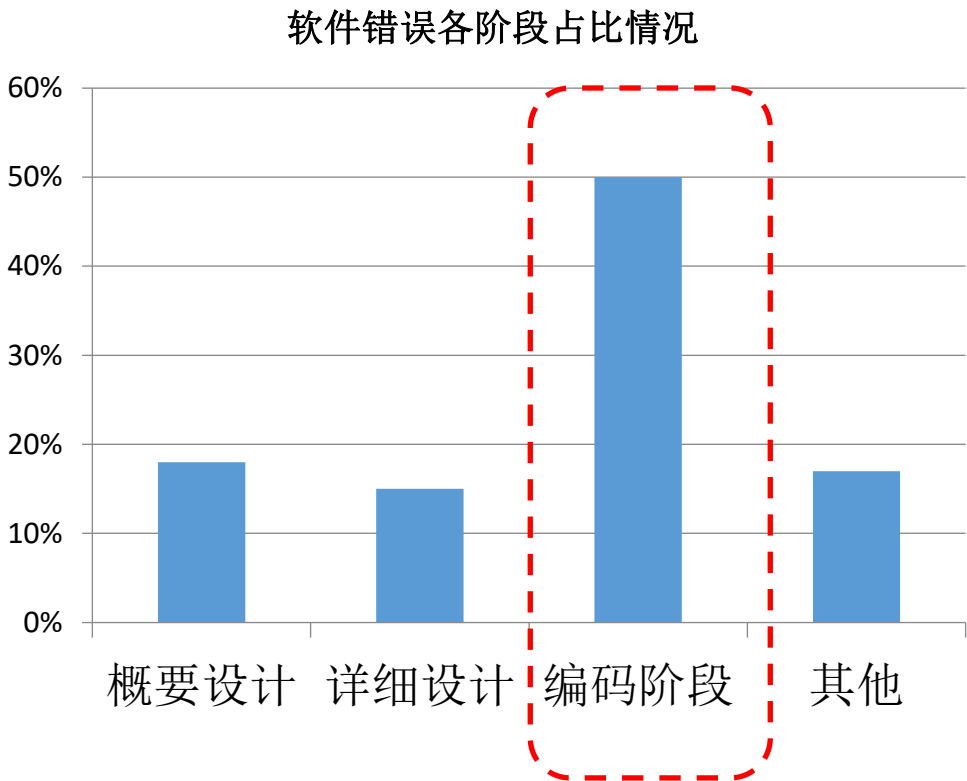
从ISO C99标准（ISO/IEC 9899:1999(E)）当中的伪随机数一节（7.20.2 Pseudo-random sequence generation functions），我们可以看到一个简明的可移植的实现样例：

```
1 static unsigned long int next = 1;
2
3 int rand(void) // RAND_MAX assumed to be 32767
4 {
5     next = next * 1103515245 + 12345;
6     return (unsigned int)(next/65536) % 32768;
7 }
8
9 void srand(unsigned int seed)
10 {
11     next = seed;
12 }
```



# 编码规范

Bell实验室的研究资料表明，软件错误中，编码阶段产生的错误占的比例接近50%。



“编码阶段” 错误种类	比例
语法错误	20%
函数(模块)之间接口错误 代码可理解度低导致优化维护阶段对代码的错误修改引起的错误 未严格检查软件逻辑导致的错误	50%以上
其他	10-30%



Bell实验室的研究人员制定了**详细的软件编程规范**，并培训每一位程序员，最终的结果把编码阶段的错误率降低至10%左右。



# 编码规范

```
#include<math.h>
#include<stdio.h>
int main()
{
    double b, c, d;
    printf( "Input b:" );
    scanf( "%lf" , &b);
    c= 4 * 3.14159 * pow(b, 2);
    d= 4.0 / 3.0 * 3.14159 * pow(b, 3);
    printf( "c=%lf\n" , c);
    printf( "d=%lf\n" , d);
    return 0;
}
```

```
/* *****
** 程序描述:用于计算球体面积及体积
** 日期:2017/4/5
** 作业题目:第一次实验操作示例
** 输入: r
** 输入范围: 正数, 大于零
** 输出: surface, volume
** surface: 球体面积, volume: 球体体积
** *****
```

```
#include<math.h>
#include<stdio.h>
#define PI 3.14159
int main()
{
    double r, surface, volume;

    //输入半径r
    printf("Input r:");
    scanf("%lf", &r);

    //输出球体表面积
    surface = 4 * PI * pow(r, 2);
    printf("surface=%lf\n", surface);

    //输出球体体积
    volume = 4.0 / 3.0 * PI * pow(r, 3);
    printf("volume=%lf\n", volume);

    return 0;
}
```

易阅读  
易测试  
易维护



# 编码规范

---

- 命名规范
- 程序版式
- 程序注释



# 编码规范-命名规范

---

## ➤ 工程命名

- 选择有实际意义的单词
- 单词与单词之间应使用下划线分开或首字符大字

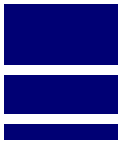
3\_1  
w111

VS

experiment3\_task1\_HeightPrediction  
Experiment3Task1HeightPrediction

Good!





# 编码规范-命名规范

## ➤ 变量命名

- 必须具有一定的实际意义;
- 命名形式为xAbcFgh, x由变量类型确定, Abc、Fgh表示连续意义字符串;
- 如果连续意义字符串仅两个, 可都大写, 如OK。

内容	类型	不规范的变量名	规范的变量名
身高	float	a, a1, temp...	fHeight
年龄	int		iAge
数学成绩	float		fMathScore

## ➤ 函数命名

- 必须具有一定的实际意义;
- 第一个字母必须使用大写字母, 要求用大小写字母组合规范函数命名, 用下划线间隔。

## ➤ 宏和常量的命名

- 必须具有一定的实际意义;
- 常量和宏定义必须全部以大写字母来撰写, 中间可根据意义的连续性用下划线连接。

```
#define PI 3.14159265
const int MAX_TIMES = 20;
```



# 编码规范-程序版式

```
float  Child_HeightCompute (float  fFaHeight,  float  fMoHeight , char cSex, char cSports, char cDiet)
{int myHeight;
if(cSex== 'm' )  myHeight=(faHeight + moHeight)*0.54;
else  myHeight = (faHeight*0.923 + moHeight)/2.0;
if(cSports== 'y' ) myHeight*=(1+0.02);
if(cDiet== 'y' ) myHeight*=0.9;
else  myHeight *=0.9;
```

```
float Child_HeightCompute (float  fFaHeight,  float  fMoHeight, char cSex, char cSports, char cDiet)
{
    → int myHeight = 0;

    if ( sex == 'm' )
        → myHeight=(faHeight + moHeight)*0.54;
    else
        myHeight = (faHeight*0.923 + moHeight)/2.0;
    →

    if ( sports == 'y' )
        myHeight*=(1+0.02);
    →

    if ( diet == 'y' )
        myHeight *= (1+0.015);
    else
        myHeight *=0.9;

}
```

缩进以 Tab 为单位；建议以下情况比上一行缩进一个Tab：

- 函数体相对函数名及{、 }
- if、 else 等之后的代码

空行：两段代码，分别完成不同的功能，这两段代码之间可使用一空行，完成逻辑上的代码区分。



- 注释的作用：是有助于对程序的阅读理解，注释语言必须**准确、易懂、简洁**。
  - 注释的数量：一般情况下，源程序有效注释量必须在**20%**以上。
  - 需要注释的位置：
    - 文件开始的位置
    - 函数开始的位置
    - 其它需要解释的地方，如常量的定义、分支语句等
-



- 函数开头的注释模板

```
/******  
** 函数名:  
** 输入: a,b,c  
** a---  
** b---  
** c---  
** 输出: x---  
** x 为 1, 表示...  
** x 为 0, 表示...  
** 功能描述:  
** 全局变量:  
** 调用模块:  
** 作者:  
** 日期:  
** 修改:  
** 日期:  
** 版本  
******/
```

- 文件开头的注释模板

```
/******  
** 文件名:  
** 创建人:  
** 日期:  
** 修改人:  
** 日期:  
** 描述:  
** 版本:  
** _____  
******/
```

- 程序中的注释模板

```
/*-----*/  
/* 注释内容 */  
/*-----*/
```

或者

```
//注释内容
```



## 实验目的

---

- 熟悉一维数组和函数指针做函数参数
  - 熟悉排序、查找、统计分析等常用算法
  - 熟悉模块化程序设计方法
  - 了解编程规范的重要性
-



## 实验内容

---

某班有最多不超过30人（具体人数由键盘输入）参加**某门**课程的考试，用**一维数组和函数指针**作函数参数，编程实现学生成绩管理系统：

- ① 录入每个学生的学号和考试成绩
  - ② 输出每个学生的学号、考试成绩（按输入的顺序输出）
  - ③ 计算并显示课程的总分和平均分
  - ④ 按成绩由高到低排出名次表
  - ⑤ 按成绩由低到高排出名次表
  - ⑥ 按学号由小到大排出成绩表
  - ⑦ 按学号查询学生排名及成绩
  - ⑧ 按优秀（90—100分）、良好（80—89分）、中等（70—79分）、及格（60—69分）、不及格（0—59分）5个类别，统计每个类别的人数以及所占的百分比
-





# 实验内容

---

要求程序运行后先显示如下菜单，并提示用户输入选项：

1. Input record
2. List record
3. Calculate total and average score of course
4. Sort in descending order by score
5. Sort in ascending order by score
6. Sort in ascending order by StudentID
7. Search by StudentID
8. Statistic analysis
0. Exit

Please enter your choice:

然后，根据用户输入的选项执行相应的操作。

**注意：除用户输入0退出程序，其余输入都不应该退出。**

---



# 实验内容

---

要求：

- 1、需要使用**函数**进行模块化程序设计；
- 2、使用**一维数组**和**函数指针**作函数参数；
- 3、程序符合本次课所讲的**编码规范**。



适当增加程序的鲁棒性，并在截图报告里进行介绍

备注：本页内容不是必做内容，供有兴趣的同学选做。

青 / 春 / 不 / 老 / 梦 / 想 / 永 / 在

FEEL THE MEANING OF THE TRIP

DREAM

MY DREAM WILL NEVER STOP

谢谢



哈爾濱工業大學(深圳)  
HARBIN INSTITUTE OF TECHNOLOGY, SHENZHEN

GO!  
TAKE YOU ON A TRIP

探索 从未停止