实验十 存储类型、链接和内存管理

10.1 存储类型实验 1

【实验内容】

加深理解存储类型。掌握各种存储类型的生命周期和作用域.

【实验目的】

通过阅读参考代码加深对存储类的理解，并理解程序原理，验证实验结果

【实验平台】

PC 机、ubuntu 操作系统，gcc 等工具

【实验步骤】

1、代码分为两个程序 parta.c 和 partb.c，程序中使用全部的 5 种存储类

2、阅读参考代码，理解代码中各种存储类的使用方法

3、参考代码如下：

partA.c

```
#include <stdio.h>

void report_count();

void accumulate(int k);

int count = 0;              // file scope, external linkage



int main(void)

{

    int value;              // automatic variable

    register int i;    // register variable



    printf("Enter a positive integer (0 to quit): ");

    while (scanf("%d", &value) == 1 && value > 0)

    {

        ++count;          // use file scope variable

        for (i = value; i >= 0; i--)
```

```
                accumulate(i);
            printf("Enter a positive integer (0 to quit): ");
        }
        report_count();

        return 0;
}

void report_count()
{
        printf("Loop executed %d times\n", count);
}
```

partB.c

```
#include <stdio.h>

extern int count;               // reference declaration, external linkage

static int total = 0;       // static definition, internal linkage
void accumulate(int k); // prototype


void accumulate(int k)    // k has block scope, no linkage
{
        static int subtotal = 0;    // static, no linkage
```

```
    if (k <= 0)
    {
        printf("loop cycle: %d\n", count);
        printf("subtotal: %d; total: %d\n", subtotal, total);
        subtotal = 0;
    }
    else
    {
        subtotal += k;
        total += k;
    }
}
```

编译：

```
gcc -o test partA.c partB.c
```

执行：

```
Enter a positive integer (0 to quit): 23
loop cycle: 1
subtotal: 276; total: 276
Enter a positive integer (0 to quit): 43
loop cycle: 2
subtotal: 946; total: 1222
Enter a positive integer (0 to quit): 0
Loop executed 2 times
```

10.2 存储类型实验 2

【实验内容】

通过编写随机数代码，加深理解存储类

【实验目的】

结合课程中所讲的随机数程序，设计一个真正的随机数产生程序

【实验平台】

PC 机、ubuntu 操作系统，gcc 等工具

【实验步骤】

1、 详细分析课上讲解的随机数程序，分析其伪随机数的特性

2、 设计一种机制，使程序能够实现用户输入种子，随后产生相应的随机数

3、 参考代码如下：

```
/* r_drive1.c -- test rand1() and srand1() */
/* compile with s_and_r.c                  */
#include <stdio.h>
extern void srand1(unsigned int x);
extern int rand1(void);

int main(void)
{
    int count;
    unsigned seed;

    printf("Please enter your choice for seed.\n");
    while (scanf("%u", &seed) == 1)
    {
        srand1(seed);       /* reset seed */
        for (count = 0; count < 5; count++)
            printf("%hd\n", rand1());
        printf("Please enter next seed (q to quit):\n");
    }
    printf("Done\n");

    return 0;
}

/* s_and_r.c -- file for rand1() and srand1()      */
```

```
/* uses ANSI C portable algorithm                    */


static unsigned long int next = 1;    /* the seed    */


int rand1(void)

{

/* magic formula to generate pseudorandom number */

        next = next * 1103515245 + 12345;

        return (unsigned int) (next/65536) % 32768;

}


void srand1(unsigned int seed)

{

        next = seed;

}
```

编译：

gcc –o test s_and_r.c r_drivel.c

执行：

./test

Please enter your choice for seed.

3

17747

7107

10365

8312

20622

Please enter next seed (q to quit):

4

1817

24166

10491

3711

15407
Please enter next seed (q to quit):
5
18655
8457
10616
31877
10193
Please enter next seed (q to quit):