

使用函数 `rand()` 生成伪随机数，使用 `srand(seed)` 重置伪随机数种子

`rand()`，无参数，返回一个范围在 0 到 `RAND_MAX` 之间的整数值。

(`#define RAND_MAX 0x7fff` (`0x7fff = 32767`))

```
/* rand0.c --生成随机数*/
/* 使用 ANSI C 可移植算法 */
static unsigned long int next = 1; /* 种子 */

unsigned int rand0(void)
{
    /* 生成伪随机数的魔术公式 */
    next = next * 1103515245 + 12345;
    return (unsigned int) (next / 65536) % 32768;
}
```

`srand(unsigned int seed)`，无返回值，用于伪随机数生成算法播种。

```
/* s_and_r.c -- 包含 rand() 和 srand() 的文件 */
/*          使用 ANSI C 可移植算法          */
static unsigned long int next = 1; /* 种子 */

int rand1(void)
{
    /*生成伪随机数的魔术公式*/
    next = next * 1103515245 + 12345;
    return (unsigned int) (next / 65536) % 32768;
}

void srand1(unsigned int seed)
{
    next = seed;
}
```

对于同一个 `seed`，每次调用 `rand()` 的返回值相同，解决如下：

使用 `time()` 函数：

C 库函数 `time_t time(time_t *seconds)` 返回自纪元 Epoch (1970-01-01 00:00:00 UTC) 起经过的时间，以秒为单位。如果 `seconds` 不为空，则返回值也存储在变量 `seconds` 中。

*声明：`time_t time(time_t *seconds)`

*参数：`seconds` -- 这是指向类型为 `time_t` 的对象的指针，用来存储 `seconds` 的值。

*返回值：以 `time_t` 对象返回当前日历时间。

(附：`time_t` 为 64 位整数)

故最终语句为：

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

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

```
srand((unsigned int)time(NULL));
```

```
rand();
```