

如何在C++中产生随机数

C++中没有自带的random函数，要实现随机数的生成就需要使用rand()和srand()。不过，由于rand()的内部实现是用线性同余法做的，所以生成的并不是真正的随机数，而是在一定范围内可看为随机的伪随机数。

- [Rand](#)
- [Srand](#)
- [通式](#)

Rand

单纯的rand()会返回一个0至RAND_MAX之间的随机数值，而RAND_MAX的值与int位数有关，最小是32767。不过rand()是一次性的，因为系统默认的随机数种子为1，只要随机数种子不变，其生成的随机数序列就不会改变。

其实，对于rand()的范围，我们可以进行人为设定的，只需要在宏定义中定义一个random(int x)函数，就可以生成范围为0至x的随机数值。当然，也可以定义为random(a,b)，使其生成范围为a至b的随机数值。具体定义方法在[通式](#)部分。

Srand

srand()可用来设置rand()产生随机数时的随机数种子。通过设置不同的种子，我们可以获取不同的随机数序列。可以利用srand((unsigned int)(time(NULL)))的方法，利用系统时钟，产生不同的随机数种子。不过要调用time()，需要加入头文件< ctime >。

示例如下：

```
#include<iostream>
#include<cstdlib>
#include<ctime>
using namespace std;
int main()
{
    srand((unsigned)time(NULL));
    for(int i=0;i<10;i++)
        cout<<rand()<<' ';
    return 0;
}
```

通式

产生一定范围随机数的通用表示公式是：

- 取得(0,x)的随机整数：rand()%x;
- 取得(a,b)的随机整数：rand()%(b-a);
- 取得[a,b)的随机整数：rand()%(b-a)+a;
- 取得[a,b]的随机整数：rand()%(b-a+1)+a;
- 取得(a,b]的随机整数：rand()%(b-a)+a+1;
- 取得0-1之间的浮点数：rand()/double(RAND_MAX)。

示例如下：

```
#include<iostream>
#include<cstdlib>
#include<ctime>
#define random(a,b) (rand()%(b-a+1)+a)
using namespace std;
```

```
int main()
{
    srand((unsigned)time(NULL));
    for(int i=0;i<10;i++)
        cout<<random(1,100)<<' ' ;
    return 0;
}
```