可能大家在编程的时候需要电脑来获取一些随机的反应，这个时候我们可以使用随机数，比较常见的是 rand() 函数，它可以随机的产生 0 ~ rand\_max 的随机数。rand\_max 是一个很大的数字，具体关系到IDE和数据类型，我们一般的需要不可能超出它的范围。  
  
下面是一个实例：

1. #include <stdio.h>
2. #include <stdlib.h>
3. int main(){
4. int a=rand();
5. printf("%d\n",a);
6. **return** 0;
7. }

编译后再运行几次，你会发现产生的随机数是相同的。实际上，rand() 函数产生的随机数是伪随机数，是根据一个数按照某个公式推算出来的，这个数我们称之为“种子”，但是这个种子在系统启动之后就是一个定值，我们需要用 srand() 来进行播种，即在int a前加一句：

1. srand((unsigned)time(NULL)); //这里利用时间进行播种，需要time.h

这样，我们就能得到不同的随机数，其实C语言中还有一个 random() 函数可以获取随机数，但是 random() 函数不是ANSI C标准，不能在VC等编译器通过，所以比较少用。  
  
那如何产生一定范围的随机数呢？我们可以利用取模的方法：

1. int a=rand()%10; //产生0~9的随机数，注意10会被整除

如果要规定上下限：

1. int a=rand()%51+13; //产生13~63的随机数

分析：取模即取余，rand()%51+13我们可以看成2部分：rand()%51是产生0~50的随机数，后面+13保证a最小只能是13，最大就是50+13=63。  
  
最后给出产生 13~63 范围内随机数的完整代码：

1. #include <stdio.h>
2. #include <stdlib.h>
3. #include <time.h>
4. int main(){
5. int a;
6. srand((unsigned)time(NULL));
7. a=rand()%51+13;
8. printf("%d\n",a);
9. **return** 0;
10. }