**第5\_2题**

*产生一组随机数，要求每个数字不能重复。例如：1，20，3，17，80，4，35，88符合要求， 3，20，1，17，80，3，35，88不符合要求*

**算法描述：**

定义randomList数列储存生成的随机数，以时间为种子值，调用rand函数生成随机数，每生成一个随机数则遍历一遍当前数组，检测到重复则重新生成随机数，若不重复，则加入数组。生成指定数量的随机数之后则遍历输出。

**源程序：**

no5\_2.c

**int** main() {  
  
 **int** randomList[MAX\_NUMBER]; //数列存储生成的随机数  
 **int** n; //需要生成的随机数个数  
 **int** min; //随机数最小值  
 **int** max; //随机数最大值  
  
 **do** {  
 puts("请输入需要生成的随机数数量：");  
 scanf("%d", &n);  
 **if**(n>=0&&n<=MAX\_NUMBER) {  
 **break**;  
 } **else**{  
 puts("请检查输入的数字，请重新输入");  
 }  
 }**while** (1);  
  
 **do** {  
 puts("请输入需要生成的随机数最小值：");  
 scanf("%d", &min);  
 puts("请输入需要生成的随机数最大值：");  
 scanf("%d", &max);  
 **if**(min<max) {  
 **break**;  
 } **else**{  
 puts("请检查输入的数字，请重新输入");  
 }  
 }**while** (1);  
 **while** (1) {  
 **for** (**int** i = 0; i < n; ++i) {  
  
 **int** newRandomNumber = 0;  
 srand((**unsigned**) time(NULL));  
 newRandomNumber = (rand() % (max - min)) + min;  
 **int** findSameFlag = 0;  
 **for** (**int** j = 0; j < i; ++j) {  
 **if** (randomList[j] == newRandomNumber) {  
 i--;  
 findSameFlag = 1;  
 **break**;  
 //检查到重复即重新生成  
 }  
 }  
 **if**(!findSameFlag)  
 randomList[i] = newRandomNumber;  
 //无重复将新随机数加入数列  
 }  
  
 puts("生成了以下随机数:");  
 **for** (**int** k = 0; k < n; ++k) {  
 printf("%d , ", randomList[k]);  
 }  
 puts("继续生成吗？(y/n)");  
 **char** choise;  
 choise = getchar();  
 **if**(choise == n){  
 **break**;  
 }  
  
 }  
  
}

**测试数据（输入、输出）：**

