|  |  |
| --- | --- |
| 1. 随机数 2. 随机数引擎和分布 3. 分布类型和引擎 4. 比较随机数引擎和rand函数 5. 引擎生成一个数值序列 6. 设置随机数发生器种子 | 函数rand  随机数引擎  Default\_random\_engine e;  E()  随机数分布类  Uniform\_int\_distribution<unsigned> u(0,9)  Default\_random\_engine e  U(e)  随机数发生器  种子 |
| 1. 随机数 2. C库的函数rand。伪随机整数，范围在0和一个系统相关的最大值 3. 问题是：只能0到一个最大值，不能指定一个范围。我们需要随机浮点、非均匀分布，   程序员为实现需引入非随机性，那么就麻烦   1. 头文件random的随机数一组协作的类：随机数引擎类、随机数分布类。 2. 引擎类是生成unsigned随机数序列，分布类是使用一个引擎类生成指定类型的、给定范围的、服从特定概率分布的随机数 3. Note：C+不应该使用库函数rand 4. 随机数引擎和分布 5. 随机数引擎是函数对象类，一个调用运算符,运算符不接受参数并返回一个随机unsigned整数，就是调用随机数引擎对象生成原始随机数 6. 表17.15. 7. 随机数引擎的输出很多不能直接使用，所以陈为原始随机数，因为生成数的范围与需要不符，转换又麻烦 8. 分布类型和引擎 9. 提供范围。分布类型也是函数对象类，也定义了一个调用运算符，接受随机数引擎作为参数。使用引擎参数生成随机数，并映射到指定的分布 10. 传递给分布对象的是引擎对象本身，因为某些分布需要调用引擎多次才能得到一个值,而不是改变的下一个值 11. Note：随机数发生器时，是指分布对象和引擎对象的组合。 12. 比较随机数引擎和rand函数 13. 类似rand输出，随机数引擎生成的unsigned整数在一个系统定义的范围内，而rand生成数的范围在0到RAND\_MAX之间。引擎类型的范围可以调用min与max成员获得 14. 引擎生成一个数值序列 15. 随机数发生器每次程序运行时序列不变，在调试有用 16. 在局部内需要设置static表明函数调用之间会保持住状态 17. 设置随机数发生器种子 18. 我们希望随机数发生器生成不同序列，可以使用种子。给引擎设置种子，引擎利用他从序列中一个新位置重新开始生成随机数 19. 两种方式，创建引擎时，调用引擎seed成员 20. 相同的种子的序列相同？待测试 21. 可以使用系统函数time作为种子拥有不同序列，但不能当程序作为一个自动过程的一部分反复运行使用time | |



