4.常用标准模块

4.1 time模块

在Python中通常有2种方式来表示时间

①时间戳，是从1970年1月1日00.00.00开始到现在的秒数

②时间元组struct\_time,其中共有九个元素。具体有:tm\_ year(年，比如2011)，tm\_mon(月),tm\_mday(日),tm\_hour(小时,0~ 23),tm\_min(分 ,0~ 59), tm\_sec (秒,0~ 59)，tm\_wday(星期，0~ 6,0表示周日),tm\_yday(一年中的第几天,1~366).tm\_isds(是否是夏令时，默认为1夏令时)。

time模块包含既有时间处理的函数，也有转换时间格式的函数。见表1所示。

表1 time 模块中的函数

函数 描述

time. asctime([tupletime]) 接受时间元组并返回一个可读的形式为"Tue Dec 11 18:07:14 2008”(2008年12月11日周二18时07分14秒)的24个字符的字符串

time. clock() 用以浮点数计算的秒数返回当前的CPU时间。用来衡量不同程序的耗时，比time. time()更有用

time. ctime([secs]) 作用相当于asctime( localtime(secs)) ,获取当前时间字符串

time. gmtime([secs]) 接收时间戳(1970纪元后经过的浮点秒数)并返回时间元组t

time. localtime([secs]) 接收时间截(1970纪元后经过的浮点秒数)并返回当地时间的时间元组t

time, mktime(tupletime) 接收时间元组并返回时间截(1970纪元后经过的浮点秒数)

time, sleep(secs) 推迟调用线程的运行,Secs指秒数

time, strftime( fmt[ , tupletime]) 接收以时间元组,并返回以可读字符申表示的当地时间，格式由fmt决定

time, strptime(str,fmt=’%a % b%d %H:%M:%S %Y’) 根据fmt的格式把一个时间字符串解析为时间元组

time. time() 返回当前时间的时间戳(1970纪元后经过的浮点秒数)

import time

print(time.localtime()) #将当前时间转换为struct\_time时间元组

#结果:time.struct\_time(tm\_year=2019, tm\_mon=1, tm\_mday=13, tm\_hour=11, tm\_min=20, tm\_sec=29, tm\_wday=6, tm\_yday=13, tm\_isdst=0)

print(time.localtime(time.time()))#将时间转化为struct\_time时间元组

#结果：time.struct\_time(tm\_year=2019, tm\_mon=1, tm\_mday=13, tm\_hour=11, tm\_min=20, tm\_sec=29, tm\_wday=6, tm\_yday=13, tm\_isdst=0)

print(time.time())#返回当前时间的时间戳，是一个浮点数

#1547349629.7957602

print(time.mktime(time.localtime()))#讲一个struct\_time转化为时间戳

结果：1547349629.0

print(time.strptime('2019-01-13 11:18:12','%Y-%m-%d %X'))#把一个格式化时间字符串转为struct\_time

结果：time.struct\_time(tm\_year=2019, tm\_mon=1, tm\_mday=13, tm\_hour=11, tm\_min=18, tm\_sec=12, tm\_wday=6, tm\_yday=13, tm\_isdst=-1)

print(time.strftime('%Y-%m-%d %X',time.localtime()))#把一个时间元组转化为格式化字符串

结果：2019-01-13 11:20:29

4.2 random模块

随机数可以用于数学、游戏等领域中，还经常被嵌人到算法中,用以提高算法效率,并提高程序的安全性。随机数函数在random模块中，常用随机数函数见表2所示。

表2random模块中的函数

函数 描 述

random. choice(seq) 从序列的元素中随机挑选一个元素,比如random.choice(range(5)),从0到94中随机挑选- -个整数

random. randrange ([start,] stop [,step]) 从指定范围内，按指定step 递增的集合中获取一个随机数,step默认值为1,比如random. randrange(5),从0到54中随机挑选一个整数

random. random() 随机生成下一个实数,它在[0,1)范围内

random. seed([x]) 改变随机数生成器的种子seed.如果你不了解其原理，你不必特别去设定seed,Python 会帮你选择seed将序列的所有元素随机排序

random. shuffle(listrandom uniform(x. y 随机生成下一个实数，它在E[x,y]范围内

4.3 math模块和cmath模块

math模块实现了许多对浮点数的数学运算函数，这些函数一般是对C语言库中同名函数的简单封装，常用的math模块的数学运算函数如表3.

math模块的数学运算函数

函数 描述

math.e 自然常数e

math.pi 圆周率pi

math. exp(x） 返回e的x次方

math. expml(x) 返回e的x次方减1

math.log(x[.base]) 返同x的以base 为底的对数. base默认为e

math. pow(x.y) 返回x的y次方

math. sqrt(x) 返回x的平方根

math. fmod(x,y) 返回x%y()取余数

math. factorial(x) 返回x阶乘

math. fabs(x) 返回x的绝对值

math. ceil( x) 返回不小于x的整数

math. floor(x) 返回不大于x的整数

math. trunc(x) 返回x对的整数部分

math. modf(x) 返回x的小树和整数

另外，在Python中cmath包含了一些用于附属运算的函数。cmath模块的函数跟math模块的函数基本一致，区别是cmath模块运算是复数，math模块运算的是数学运算。

import cmath

print(cmath.sqrt(-1))

#结果：1j

print(cmath.sqrt(9))

#结果：(3+0j)