

In [1]:

```
1 from math import log, exp, sqrt, pow
2
```

2.8

In [2]:

```
1 import sys
2 from prettytable import PrettyTable
```

In [3]:

```
1
2
3 result=0
4 result1 = 0
5 result2 = 0
6 result3=0
7 a=0
8 b=0
9 c=0
10 d=0
11
12 for i in range(16):
13     if i ==0:
14         result = 200*2**i
15         a=i
16     if i ==5:
17         result1 = 200*2**i
18         b=i
19     if i ==10:
20         result2 = 200*2**i
21         c=i
22     if i ==15:
23         result3 = 200*2**i
24         d=i
25
26 table = PrettyTable()
27
28 table.add_column("hour", [a,b,c,d])
29 table.add_column("Number of Bacteria", [result,result1,result2,result3])
30 table.align["Number of Bacteria"] = 'l'
31 #table.align["Number of Bacteria"] = 'r'
32 #table.align["hour"] = 'l'
33 print(table)
34
```

hour	Number of Bacteria
0	200
5	6400
10	204800
15	6553600

In [4]:

```
1
2
3 result=0
4 result1 = 0
5 result2 = 0
6 result3=0
7 a=0
8 b=0
9 c=0
10 d=0
11
12 for i in range(16):
13     if i ==0:
14         result = 200*2**i
15         a=i
16     if i ==5:
17         result1 = 200*2**i
18         b=i
19     if i ==10:
20         result2 = 200*2**i
21         c=i
22     if i ==15:
23         result3 = 200*2**i
24         d=i
25
26 table = PrettyTable()
27
28 table.add_column("hour", [a,b,c,d])
29 table.add_column("Number of Bacteria", [result,result1,result2,result3])
30
31 table.align["Number of Bacteria"] = 'r'
32 table.align["hour"] = 'r'
33 print(table)
34
```

hour	Number of Bacteria
0	200
5	6400
10	204800
15	6553600

In [5]:

```
1 o=10
2 n=5
3 n1=2
4 d=0
5 w=0
6 w1=0
7 p=0.03
8 p1=-0.03
9 y=0
10 w=o*(1+p)**n
11 w1=w*(1+p1)**n1
12 print(w1)
13
14
```

10.907609765088699

2.13

In Python 2, it has two kinds of integers: one is short integer, which is often referred to as integer, which is represented by int and has a finite size, and a long integer with infinite size. Python 3 integrates two integer representations. In theory, Python integers have no upper limit, as long as they do not exceed the memory space.

In [16]:

```
1 for i in range(100,9000):
2     print(2**i)
```

```
1267650600228229401496703205376
2535301200456458802993406410752
5070602400912917605986812821504
10141204801825835211973625643008
20282409603651670423947251286016
40564819207303340847894502572032
81129638414606681695789005144064
162259276829213363391578010288128
324518553658426726783156020576256
649037107316853453566312041152512
1298074214633706907132624082305024
2596148429267413814265248164610048
5192296858534827628530496329220096
10384593717069655257060992658440192
20769187434139310514121985316880384
41538374868278621028243970633760768
83076749736557242056487941267521536
166153499473114484112975882535043072
332306998946228968225951765070086144
664613997898457688451008580140176000
```

In []:

```
1
```

