你跟一个人比赛，你第i局获胜的概率为1/i，第i局获胜所得money是ai，输了不赔钱，求n局后赢的money数的期望

ai<=1e9

n<=1e5

Input:

第一行输入T表示case数 ,T后面是一行空行

对于每组样例：第一行输入n

接下来一行输入a1到an

Output：

输出期望 ,保留10位小数

样例1输入：

1

3

1 2 3

样例1输出：

3.0000000000

题目很简单，就是扫一遍ans+=a[i]/i

但是有个问题

浮点数在存储的时候，是一部分位存数字值，一部分位存1e几什么的就是控制小数点的位数

而当你给一个Long double类型的变量赋值的时候，如果赋的值的大小超过了long double用来存数字的这些位的最大值，他就是把大的截去，就会产生误差

这样的话，如果一组数据是

10000000 10000000……..

2 2 2……….

10000000 10000000……..

当计算到2之前和就已经达到变量能存的值的上限了，这样你再加一个2就有可能会被截取了，就是丢掉不管，而这样的话，如果2的数量很多很多，那么这些2的和就是一个很大的数值，而这个数值被舍去了