无标题笔记

https://atcoder.jp/contests/abc326/tasks/abc326_e

所以我们得到最终表格：

	1	2	3	4
第 1 次	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
第 2 次	0	$\frac{1}{4^2}$	$\frac{2}{4^2}$	$\frac{3}{4^2}$
第 3 次	0	0	$\frac{1}{4^3}$	$\frac{3}{4^3}$
第 4 次	0	0	0	$\frac{1}{4^4}$

发现了什么？第 j 次贡献的分母是 n^j，而对于每个数 i 来说，其分子依次为杨辉三角的第 i 横排。所以最终所求答案即为：

$$\sum_{i=1}^{n}\sum_{j=1}^{i}\frac{C_{i-1}^{j-1}}{n^j}\times a_i$$

我们设第 i 个数的贡献为 f_i，然后根据杨辉三角的优秀性质，就可以得到递推式：

$$f_i = f_{i-1} + f_{i-1} \times \frac{1}{n}$$

然后问题就解决了。

```
#include <bits/stdc++.h>
using namespace std;
#define itn int
#define icn cin
#define int long long
#define endl '\n'
typedef pair<int,int> PII;
```

```
const int N=3e5+10;
const int INF=1e18;
const int mod=998244353;
#define ll __int128

int a[N],f[N];
int ans;
int qmi(int a,int b)
{
  int t=1;
  while(b!=0)
  {
   if(b&1)t=t*a%mod;
   b>>=1;
   a=a*a%mod;
  }
  return t;
}
void solve()
{
   itn n;cin>>n;
   for(itn i=1;i<=n;i++)
   cin>>a[i];
   int inv=qmi(n,mod-2);
   f[1]=inv;
   ans+=f[1]*a[1]%mod;
   for(int i=2;i<=n;i++)
   {
```

```
        f[i]=f[i-1]+f[i-1]*inv%mod,f[i]%=mod;
        ans+=f[i]*a[i]%mod;
        ans%=mod;
    }
        cout<<ans;
}
signed main(){
cin.tie(0)->sync_with_stdio(0);
    int T=1;
//    cin>>T;
    while(T--)solve();
return 0;
}
```