

技巧

优读

```
1 #include<iostream>
2 #include<cstring>
3 #include<cstdio>
4 using namespace std;
5 int n,a[1010];
6 int inline read()
7 {
8     int num=0;
9     char c;
10    bool plus=true;
11    while((c=getchar())==' '||c=='\n'||c=='\r');
12    if(c=='-') plus=false;
13    else num=c-'0';
14    while(isdigit(c=getchar()))
15        num=num*10+c-'0';
16    return num*(plus?1:-1);
17 }
18 int main()
19 {
20     n=read();
21     for(int i=1;i<=n;i++)
22         a[i]=read();
23     printf("%d\n",n);
24     for(int i=1;i<=n;i++)
25         printf("%d ",a[i]);
26     printf("\n");
27     return 0;
28 }
```

斐波那契通项公式

```
1 #include<iostream>
2 #include<cstdio>
3 #include<cmath>
4 using namespace std;
5 long double n=0;
6 //int p; printf("%lld ",(long long)((pow(((1+sqrt(5))/2),n)-pow(((1-sqrt(5))/2),n))/sqrt(5)));
7 //f[2*n]=f[2*n-1]+...+f[3]+f[1];
8 //f[2*n+1]-1=f[2*n]+...+f[4]+f[2];
9 //f[2*n]=f[n]*f[n]+2*f[n]*f[n-1];
10 int main(){
11     //scanf("%d",&p);
12     for(int i=1;i<=92;i++){
13         n++;
14         printf("%lld ",(long long)((pow(((1+sqrt(5))/2),n)-pow(((1-sqrt(5))/2),n))/sqrt(5)));
15         if(i%5==0) cout<<endl;
```

```
16     }
17     return 0;
18 }
```

斐波那契公约数

```
1 //luogu1306
2 #include<iostream>
3 #include<cstdio>
4 #include<cstring>
5 using namespace std;
6 long long n,m,c,d,p=100000000;
7 struct jz{
8     long long a[40][40];
9 };
10 long long gcd(long long a,long long b){
11     return b==0?a:gcd(b,a%b);
12 }
13 jz s;
14 jz fz(jz x){
15     memset(x.a,0,sizeof(x.a));
16     for(long long i=1;i<=n;i++){
17         x.a[i][i]=1;
18     }
19     return x;
20 }
21 jz cf(jz x,jz y){
22     jz neww;
23     memset(neww.a,0,sizeof(neww.a));
24     for(long long i=1;i<=n;i++){
25         for(long long j=1;j<=n;j++){
26             for(long long k=1;k<=n;k++){
27                 neww.a[i][j]=((x.a[i][k]*y.a[k][j])%p+neww.a[i][j])%p;
28             }
29         }
30     }
31     return neww;
32 }
33 jz speed(jz x,long long b){
34     if(b==1) return x;
35     jz cur,ans;
36     ans=fz(ans);
37     cur=x;
38     while(b){
39         if(b&1){
40             ans=cf(ans,cur);
41         }
42         cur=cf(cur,cur);
43     }
44     b>>=1;
45 }
46 return ans;
```

```

47 }
48 int main(){
49
50     s.a[1][1]=1;
51     s.a[1][2]=1;
52     s.a[2][1]=1;
53     s.a[2][2]=0;
54     n=2;
55     //scanf("%d",&m);
56     scanf("%lld%lld",&c,&d);
57     long long e=gcd(c,d);
58     jz x=speed(s,e-1);
59     printf("%lld\n",x.a[1][1]%p);
60     return 0;
61 }

```

斐波那契矩阵快速幂

```

1  #include<iostream>
2  #include<cstdio>
3  #include<cstring>
4  using namespace std;
5  int n,m;
6  struct jz{
7      int a[40][40];
8  };
9  jz s;
10 jz fz(jz x){
11     memset(x.a,0,sizeof(x.a));
12     for(int i=1;i<=n;i++){
13         x.a[i][i]=1;
14     }
15     return x;
16 }
17 jz cf(jz x,jz y){
18     jz neww;
19     memset(neww.a,0,sizeof(neww.a));
20     for(int i=1;i<=n;i++){
21         for(int j=1;j<=n;j++){
22             for(int k=1;k<=n;k++){
23                 neww.a[i][j]+=x.a[i][k]*y.a[k][j];
24             }
25         }
26     }
27     return neww;
28 }
29 jz speed(jz x,int b){
30     if(b==1) return x;
31     jz cur,ans;
32     ans=fz(ans);
33     cur=x;
34     while(b){
35         if(b&1){

```

```

36         ans=cf(ans,cur);
37     }
38     cur=cf(cur,cur);
39
40     b>>=1;
41 }
42 return ans;
43 }
44 int main(){
45
46     s.a[1][1]=1;
47     s.a[1][2]=1;
48     s.a[2][1]=1;
49     s.a[2][2]=0;
50     n=2;
51     scanf("%d",&m);
52     jz x=speed(s,m-1);
53     printf("%d\n",x.a[1][1]);
54     return 0;
55 }

```

接近分组

```

1  #include<bits/stdc++.h>
2  #define ll long long
3  #define db double
4  #define ld long double
5  using namespace std;
6  int n,m;
7  int main(){
8      //cut n to m
9      cin>>n>>m;
10     for(int i=1;i<=m;i++){
11         if(i!=1) printf(" ");
12         cout<<ceil((n-i+1)/(db)m);
13     }
14     cout<<endl;
15     return 0;
16 }

```