```
namespace FFT {
  const int mod = 998244353;
  const int LOG = 20;
  const int MAX = 1<<LOG;
  int W;
  int Wrev;
  int rev[MAX];
  int wp[MAX];
  int F[2][MAX];
  int bpow(int base, int exp , int mo = mod) {
    int res = 1;
    for(; exp; exp >>= 1, base = 1ll * base * base % mo)
        if(exp & 1) res = 1ll & res * base  % mo;
    return res;
  }

  void init() {
    W = 2;
    for ( ;; ) {
      int cc = W;
      for(int t = 1; t < LOG; t++) cc = 1LL * cc * cc % mod;
      if(cc == mod - 1) break;
      W++;
    }
    Wrev = bpow(W, mod - 2, mod);
    wp[0] = 1;
    for(int i = 1; i < MAX; i++) wp[i] = 1LL * wp[i-1] * W % mod;
    for(int mask = 1;mask < MAX;mask ++ ) {
      rev[mask] = rev[mask ^ (mask & -mask)] ^ (1 << (LOG - 1 -
       __builtin_ctz(mask)));
    }
  }

  void fft(vector<long long> & x, int lev) {
    int L = 1 << lev;
    for( int i = 0;i<L;i++ ) F[0][rev[i] >> ( LOG - lev ) ] = x[i];
    int c = 0, cc = 1;
    for(int l = 0; l < lev; l++) {
      int len = 1 << l;
      for(int st = 0; st < L; st += (len << 1))
        for(int i = 0;i<len;i++)
        {
          long long xx = 1LL * F[c][st + len + i] * wp[i << (LOG - 1 - l)] %
           mod;
          F[cc][st + i] = (xx + F[c][st + i]) % mod;
          F[cc][st + len + i] = (F[c][st + i] - xx + mod) % mod;
        }
      swap(c, cc);
    }
    for(int i = 0; i < L; i++) x[i] = F[c][i];
  }

  vector<long long> mul(vector<long long > &a, vector<long long> &b) {
    int sz = 1,xx = 0;
    while(sz < max(a.size() , b.size())) sz <<= 1, xx++;
    sz <<= 1, xx++;
    a.resize(sz), b.resize(sz);
    vector<long long> res = vector<long long>(sz, 0);
    fft(a, xx), fft (b, xx);
    for(int i = 0; i < sz; i++) a[i] = (a[i] * b[i]) % mod;
    fft(a, xx);
    long long revN = 1;
    for(int i = 0; i < xx; i++) {
      if(revN & 1) revN += mod;
      revN /= 2;
    }
    for(int i = 0; i < sz; i++)
      res[i] = (a[i] * revN) % mod;
    reverse(res.begin() +1,res.end());
    return res;
  }
};

// wthtxdy
#define rep(i,a,n) for (int i=a;i<n;i++)
#define per(i,a,n) for (int i=n-1;i>=a;i--)
#define pb push_back
#define mp make_pair
#define all(x) (x).begin(),(x).end()
#define SZ(x) ((int)(x).size())
#define fi first
#define se second
typedef vector<int> VI;
typedef long long ll;
typedef pair<int,int> PII;
const ll mod=1000000007;
ll powmod(ll a,ll b) {ll
 res=1;a%=mod;for(;b;b>>=1){if(b&1)res=res*a%mod;a=a*a%mod;}return res;}
// head

const int N=201000;
int mth[N][5],s[N],t[N],n,m,k,mv[N],cnt;
char S[N],T[N];
set<int> st[5];
VI v1,v2,v3;

typedef complex<double> CD;
const double pi=acos(-1.0);
namespace fft{
    const int N=1<<19;
    CD cp1[N+10],cp2[N+10];
    int R[N+10];
    void rev(CD* a,int n) { for (int i=0;i<n;i++) if (R[i]<i)
     swap(a[R[i]],a[i]);}
    void dft(CD* a,int n,bool inv) {
        CD wi,w,u,v;
        rev(a,n);
        for (int m=2;m<=n;m<<=1) {
            double Arg=2*pi/m*(inv?-1:1);
            wi=CD(cos(Arg),sin(Arg));
            for (int j=0;j<n;j+=m) { w=1;
                for (int k=j,k2=j+m/2;k2<j+m;k++,k2++)
                    u=a[k],v=a[k2]*w,a[k]=u+v,a[k2]=u-v,w=w*wi;
            }
        }
        if (inv) for (int j=0;j<n;j++) a[j]=a[j]/(1.*n);
    }
    VI solve(VI& p1,VI& p2) {
        int n=p1.size()+p2.size()+1;
        int l=1,cnt=0;
        while (l<=n) l+=l,cnt++;
        rep(i,0,l) { R[i]=0;rep(j,0,cnt) R[i]=(R[i]<<1)|((i>>j)&1);}
        rep(i,0,l) cp1[i]=0,cp2[i]=0;
        rep(i,0,SZ(p1)) cp1[i]=p1[i];
        rep(i,0,SZ(p2)) cp2[i]=p2[i];
        dft(cp1,l,0);
        dft(cp2,l,0);
        rep(i,0,l) cp1[i]*=cp2[i];
        dft(cp1,l,1);
        VI res;
        res.clear();
        n=p1.size()+p2.size()-1;
        rep(i,0,n) res.pb((ll)floor(cp1[i].real()+0.5));
        return res;
    }
}
```