```cpp
#include<bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef pair<int, int> pii;
#define fi first
#define se second
#define empb emplace_back
#define all(x) (x).begin(),(x).end()
const int N = 1e5 + 100;
const int mod = 1e9 + 7;
bitset<N> np;
int m=0, B;
int _id[N*2];
ll p[N], sp[N], g1[N*2], g2[N*2], w[N*2], n;
void init(int B) {
    np.reset(); p[0] = 0;
    for(int i = 2; i <= B; i++) {
        if(!np[i]) p[++p[0]] = i, sp[p[0]] = sp[p[0]-1] + i;
        for(int j = 1; j <= p[0] && i * p[j] <= B; j++) {
            np[i * p[j]] = 1;
            if(i % p[j] == 0) break;
        }
    }
}
int id(ll x) { return x <= B? _id[x] : _id[n/x+B];}
ll fpk(ll p, ll e, ll pe) {
    return pe / p * (p - 1) % mod;
}
ll S(ll x, int y) {
    if(x<=1||p[y]>x) return 0;
    int k = id(x);
    ll res = (g2[k]-g1[k]-sp[y-1]+y-1)%mod;
    for(int i = y; i<=p[0]&&p[i]*p[i]<=x; i++) {
        ll p1 = p[i], p2 = p[i]*p[i];
        for(int e = 1; p2 <= x; e++,p1=p2,p2*=p[i])
            res=(res+S(x/p1,i+1)*fpk(p[i],e,p1)+fpk(p[i],e+1,p2))%mod;
    }
    if(res < 0) res += mod;
    return res;
}
int main() {
#ifdef local
    freopen("in.txt", "r", stdin);
#endif
    ios::sync_with_stdio(false);
    cin.tie(0), cout.tie(0);
    while(cin >> n) {
        B = sqrt(n);
        init(B); m = 0;
        for(ll l = 1, r; l <= n; l = r + 1) {
            r = n / (n / l);
            w[++m] = n / l;
            _id[w[m]<=B?w[m]:n/w[m]+B] = m;
            g1[m] = (w[m] - 1) % mod;
            g2[m] = (w[m] + 2) % mod * g1[m] % mod;
            if(g2[m] & 1) g2[m] += mod; g2[m] /= 2;
        }
        for(int j = 1; j <= p[0]; j++) {
            for(int i = 1; i <= m && p[j]*p[j]<=w[i]; i++) {
                int k = id(w[i]/p[j]);
                g1[i] = (g1[i] - g1[k] + j - 1) % mod;
                g2[i] = (g2[i] - p[j]*(g2[k] - sp[j-1])) % mod;
            }
        }
        for(int i = 1; i <= m; i++) {
            if(g1[i] < 0) g1[i] += mod;
            if(g2[i] < 0) g2[i] += mod;
        }
        cout << S(n, 1) + 1 << '\n';
    }
```

```
71      return 0;
72  }
```