```
ModularInteger operator - (long long x) {
        x %= MOD;
        if (x < 0) x += MOD;
        ModularInteger res;
        res.val = val - x;
        if (res.val < 0) res.val += MOD;
        return res;
}

ModularInteger operator * (long long x) {
        x %= MOD;
        if (x < 0) x += MOD;
        ModularInteger res;
        res.val = val * x;
        res.val %= MOD;
        return res;
}

void operator += (long long x) {
        x %= MOD;
        if (x < 0) x += MOD;
        val = val + x;
        if (val >= MOD) val -= MOD;
}

void operator -= (long long x) {
        x %= MOD;
        if (x < 0) x += MOD;
        val = val - x;
        if (val < 0) val += MOD;
}

void operator *= (long long x) {
        x %= MOD;
        if (x < 0) x += MOD;
        val = val * x;
        val %= MOD;
}
```