

拉格朗日插值

题目背景

这是一道模板题

题目描述

由小学知识可知 n 个点 (x_i, y_i) 可以唯一地确定一个多项式 $y = f(x)$

现在，给定这 n 个点，请你确定这个多项式，并求出 $f(k) \bmod 998244353$ 的值

输入格式

第一行两个整数 n, k

接下来 n 行，第 i 行两个整数 x_i, y_i

输出格式

一行一个整数，表示 $f(k) \bmod 998244353$ 的值。

输入输出样例

输入 #1

```
3 100
1 4
2 9
3 16
```

输出 #1

```
10201
```

输入 #2

```
3 100
1 1
2 2
3 3
```

输出 #2

100

说明/提示

样例一中的多项式为 $f(x)=x^2+2x+1$ $f(100) = 10201$

样例二中的多项式为 $f(x)=x$ $f(100) = 100$

$1 \leq n \leq 2 \times 10^3$ $1 \leq x_i, y_i, k < 998244353$ x_i 两两不同

拉格朗日插值的公式大概是 $f(k) = \sum_{i=0}^n y_i \prod_{j \neq i} \frac{k - x_j}{x_i - x_j}$ x_i, y_i 是在 x_i 的取值。

```
#include <bits/stdc++.h>
#define int long long
using namespace std;
struct io {
    char buf[1 << 26 | 3], *s; int f;
    io() { f = 0, buf[fread(s = buf, 1, 1 << 26, stdin)] = '\n'; }
    io& operator >> (int&x) {
        for(x = f = 0; !isdigit(*s); ++s) f |= *s == '-';
        while(isdigit(*s)) x = x * 10 + (*s++ ^ 48);
        return x = f ? -x : x, *this;
    }
};

const int mod = 998244353;
int qpow(int x, int y) {
    int res = 1;
    for(; y; y >>= 1, x = x * x % mod)
        if(y & 1) res = res * x % mod;
    return res;
}
int inv(int x) { return qpow(x, mod - 2); }

int n, k;
const int maxn = 2e3 + 32;
int x[maxn], y[maxn];

#define out cout
signed main() {
#ifdef LOCAL
#define in cin
    ios :: sync_with_stdio(false), cin.tie(nullptr), cout.tie(nullptr);
    freopen("testdata.in", "r", stdin);
#else
    io in;
#endif
}
```

```
#endif
in >> n >> k;
for(int i = 1 ; i <= n ; i ++ ) in >> x[i] >> y[i];
int ans = 0;
for(int i = 1 ; i <= n ; i ++ ) {
    int a, b; a = b = 1;
    for(int j = 1 ; j <= n ; j ++ ) if(i ^ j) { a = a * (k - x[j]) % mod; }
    for(int j = 1 ; j <= n ; j ++ ) if(i ^ j) { b = b * (x[i] - x[j]) % mod; }
}
a = (a + mod) % mod, b = (b + mod) % mod, b = inv(b);
ans = (ans + a * b % mod * y[i] % mod) % mod;
}
out << ans << '\n';
return 0;
}
```

!= → \neq

From:
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:
<https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:manespace:%E6%8B%89%E6%A0%BC%E6%9C%97%E6%97%A5%E6%8F%92%E5%80%BC%E6%B3%95&rev=1589001602>

Last update: 2020/05/09 13:20