Butterfly effect

1416 64% 30 ★★★½★ 8 votes

Applications of Dynamic Programming, Algorithms, Dynamic programming, Dynamic Programming

Share

N/A

Author's Solution

```
#include <bits/stdc++.h>
#define ll long long
using namespace std;

const ll MAX = 1e5 + 5, MOD = 1e9 + 7;

ll dp[MAX][10];

ll solve(ll i, ll p, ll &n, ll &k, ll &x){
    if(i == n)return 1;

    ll &ans = dp[i][p];
    if(ans != -1)return ans;
    ans = 0;

    for(int j = 1; j <= k; j++){

        if(j != p){
            ans += solve(i + 1, j, n, k, x);
            ans %= MOD;
        }
    }

    if(p == x)ans += solve(i + 1, p, n, k, x);

    return ans;
}
int main()
{

//    freopen("in.txt", "r", stdin);
//    freopen("out.txt", "w", stdout);

    ll t;
    cin >> t;
    while(t--){
        memset(&dp, -1, sizeof dp);
        ll n, k, x;
        cin >> n >> k >> x;

        ll ans = 0;
        for(int i = 1; i <= k; i++){

            ans += solve(1, i, n, k, x);
            ans %= MOD;
        }

        cout << ans << endl;


    }


    return 0;
}
```

Tester's Solution

```
#include<bits/stdc++.h>
#define Mod 1000000007
#define lld long long
using namespace std;

lld dp[100010];
int main()
{
        //freopen("input.txt","r",stdin);
        //freopen("output.txt","w",stdout);
        int t;
        cin>> t;

        while(t-- )
        {
                int n,x,k;

                cin>> n>>k>>x;
                for(int i = 1; i<= n; i++ ) dp[i]=0;
                dp[0] =1;
                dp[1] = k; // only one point on x-axis k- ways to color them
                lld val = 1;
                for(int i = 2; i<= n; i++ )
                {
                        dp[i] = (dp[i-1]*1LL* (k-1)) %Mod;
                        if(x<= k) dp[i]= (dp[i]+dp[i-2])%Mod;
                }
                cout<<dp[n] <<endl;
        }

        return 0;
}
```

?