

```

struct DSU_WITH_ROLLBACK {
    vector<int> rnk;
    int comps;
    struct dsu_save {
        int u, urank, v, vrank, comps;
    };
    stack<dsu_save> op;

    DSU_WITH_ROLLBACK() {
    };

    DSU_WITH_ROLLBACK(int n) {
        Init(n);
    };

    void Init(int n) {
        rnk.clear();
        while (!op.empty())
            op.pop();
        rnk.resize(n+1);
        for(int i=1; i<=n; ++i) {
            rnk[i]=1;
        }
        comps=n;
    }

    int find(int x) {
        if (rnk[x]>0)
            return x;
        int y=find(-rnk[x]);
        return y;
    }

    void join(int a,int b, int save=0) {
        int ra=find(a), rb=find(b);
        if (save)
            op.push({ra,rnk[ra],rb,rnk[rb],comps});
        if (ra==rb)
            return;
        comps--;
        if (rnk[ra]<rnk[rb])
            swap(ra, rb);
        rnk[ra] += rnk[rb];
        rnk[rb] = -ra;
    }

```

```
}

void rollback() {
    if (op.empty())
        return;
    dsu_save x=op.top();
    op.pop();
    comps=x.comps;
    rnk[x.u]=x.urank;
    rnk[x.v]=x.vrank;
}

void panic() {
    while (!op.empty())
        rollback();
}

};
```