最小生成树K

K较为简便，P直接修改Dij即可

数据、并查集

```
class ufs {
public:
    ufs(int n)
    {
        fa = new int[n + 1];
        for (int i = 1; i <= n; ++i)
            fa[i] = i;
        block = n;
    }
    void uni(int i, int j)
    {
        int fi = find(i);
        block = fi == find(j) ? block : block - 1;
        fa[find(i)] = fi;
    }
    int find(int x)
        return x == fa[x] ? x : (fa[x] = find(fa[x]));
    bool same(int i, int j)
        return find(i) == find(j);
    int blocks() { return block; }
private:
    int block;
    int* fa;
};

int edgnum = 0;
edge edg[MAXN];
```

本体

```
void addedg_Kru(int a, int b, llint w)
{
    edg[edgnum++] = edge(a, b, w);
}

llint Kruskal()
{
    ufs set = ufs(n);
    llint ans = 0;
    sort(edg, edg + m);
    for (int i = 0; i < m; i++)
    {
```

```
            if (!set.same(edg[i].u, edg[i].v))
            {
                set.uni(edg[i].u, edg[i].v);
                //可以在此记录所用的边
                ans += edg[i].w;
            }
        }
        return ans;
    }
```