Bellman_ford

```cpp
#include <bits/stdc++.h>
using namespace std;

const int N = 1e5 + 5, M = 5e5 * 2 + 10;
int n, m, s, t, k;
int dist[N], backup[N];
bool vis[N];
struct edge{
    int u, v, w;
}e[M];

//  最多经过k条边的最短路
int bellman_ford(){
    memset(dist, 0x3f, sizeof(dist));
    dist[s] = 0;
    for(int p = 0; p < k; p++){
        memcpy(backup, dist, sizeof(dist));
        for(int i = 0; i < m; i++){
            int u = e[i].u, v = e[i].v, w = e[i].w;
            dist[v] = min(dist[v], backup[u] + w);
        }
    }

    if(dist[t] > 0x3f3f3f3f / 2) return -1;
    else return dist[t];
}

int main(){
    cin >> n >> m >> s >> t >> k;
    for(int i = 0; i < m; i++){
        int u, v, w;
        scanf("%d%d%d", &e[i].u, &e[i].v, &e[i].w);
    }

    int t = bellman_ford();

    if(t == -1) puts("NO");
    else printf("%d\n", t);
    return 0;
}
```