

```
1 #include "../bits/stdc++.h"
2 #include "../graph.hpp"
3
4 // 単一始点最短経路 O(EV)
5 // 負閉路があれば true, otherwise false
6 bool findNegativeLoop(int n, const std::vector<Edge> es, int s, std::vector<int> &d)
7 {
8     int m = es.size();
9     std::vector<int> pre(n, -1);
10    std::vector<bool> negative(n, false);
11    d[s] = 0;
12    for (int j = 0; j < n - 1; j++)
13    {
14        for (int i = 0; i < m; i++)
15        {
16            Edge e = es[i];
17            if (d[e.from] != INF && d[e.to] > d[e.from] + e.cost)
18            {
19                d[e.to] = d[e.from] + e.cost;
20                pre[e.to] = pre[e.from];
21            }
22        }
23    }
24    for (int i = 0; i < m; i++)
25    {
26        Edge e = es[i];
27        if (d[e.from] != INF && d[e.from] + e.cost < d[e.to])
28        {
29            return true;
30        }
31    }
32    return false;
33 }
34
```