

```
1 #include "../bits/stdc++.h"
2 #include "../graph.hpp"
3
4 // verified: http://judge.u-aizu.ac.jp/onlinejudge/review.jsp?rid=3381541
5
6 // プリム法による最小全域木
7 // 到達可能な頂点への辺の内, コストが最小のものを選んでいく
8 //  $O(E \log V)$ 
9 int prim(const Graph &g)
10 {
11     int sz = g.size();
12     int ret = 0;
13     std::vector<bool> used(sz, false);
14     std::priority_queue<P, std::vector<P>, std::greater<P>> que;
15     que.push(P(0, 0));
16     while (!que.empty())
17     {
18         P p = que.top();
19         que.pop();
20         if (used[p.second])
21             continue;
22         used[p.second] = true;
23         ret += p.first;
24         for (const auto &e : g[p.second])
25         {
26             que.push(P(e.cost, e.to));
27         }
28     }
29     return ret;
30 }
31
```