a

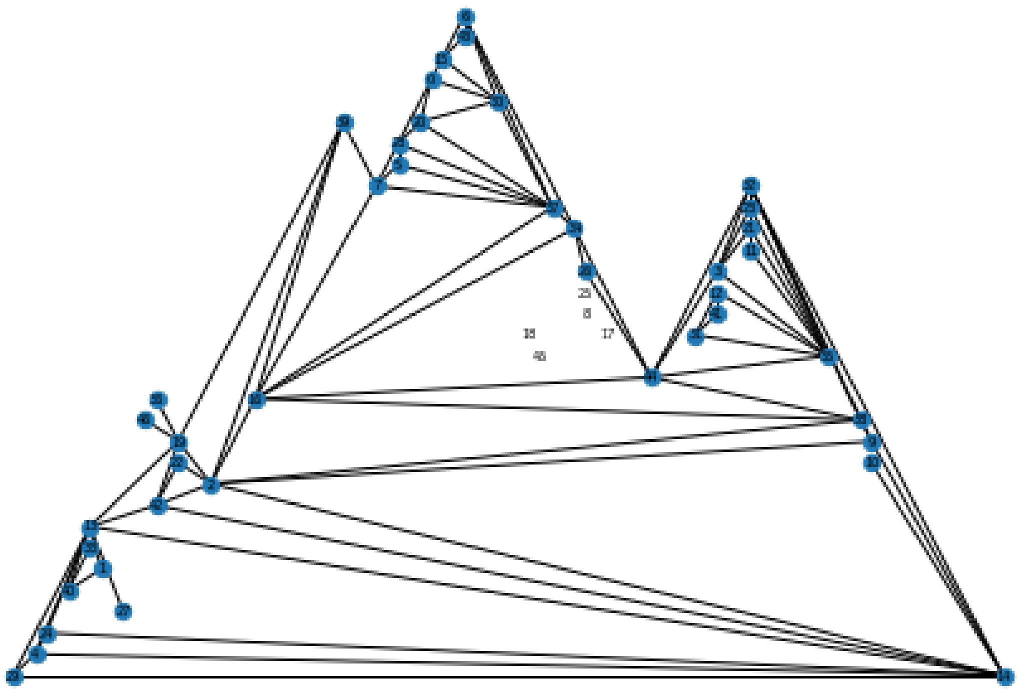
Statement

Prove that G' is a planar by drawing it on a plane with no intersection of edges.

Answer

```
0: Albania
1: Andorra
2: Austria
3: Belarus
4: Belgium
5: Bosnia and Herzegovina
6: Bulgaria
7: Croatia
8: Cyprus
9: Czech Republic
10: Denmark
11: Estonia
12: Finland
13: France
14: Germany
15: Greece
16: Hungary
17: Iceland
18: Ireland
19: Italy
20: Kosovo
21: Latvia
22: Liechtenstein
23: Lithuania
24: Luxembourg
25: Malta
26: Moldova
27: Monaco
28: Montenegro
29: Netherlands
30: North Macedonia
31: Norway
32: Poland
33: Portugal
34: Romania
35: Russia
36: San Marino
37: Serbia
38: Slovakia
39: Slovenia
```

```
40: Spain
41: Sweden
42: Switzerland
43: Turkey
44: Ukraine
45: United Kingdom
46: Vatican City
```



b

Statement

Find $|V|$, $|E|$, $\delta(G)$, $\Delta(G)$, $rad(G)$, $diam(G)$, $girth(G)$, $center(G)$, $\kappa(G)$, $\lambda(G)$.

Definitions

$|V|$ is the number of vertices in G

$|E|$ is the number of edges in G

$\delta(G) = \min_{v \in V} deg(v)$

$\Delta(G) = \max_{v \in V} deg(V)$

$\varepsilon(v) = \max_{u \in V} dist(v, u)$

$rad(G) = \min_{v \ inV} \varepsilon(v)$

$diam(G) = \max_{v \in V} \varepsilon(v)$

$girth(G)$ is the length of the shortest cycle in the G

$center(G) = \{v \in V \mid \varepsilon(v) = rad(G)\}$

$\kappa(G)$ is the minimum number of vertices, the removal of which would result in the number of connected components in G to be greater than 1

$\lambda(G)$ is the minimum number of edges, the removal of which would result in the number of connected components in G to be greater than 1

Answer

```
|V| = 42
|E| = 88
δ(G) = 1
Δ(G) = 9
rad(G) = 5
diam(G) = 8
girth(G) = 3
center(G) = ['Switzerland', 'Slovakia', 'Germany', 'Belarus', 'Czech Republic',
'Poland', 'Slovenia', 'Russia', 'Croatia', 'Austria', 'Ukraine', 'Hungary']
κ(G) = 1
λ(G) = 1
```

c

Statement

Find the minimum vertex coloring $Z : V \to \mathbb{N}$ of G

Definitions

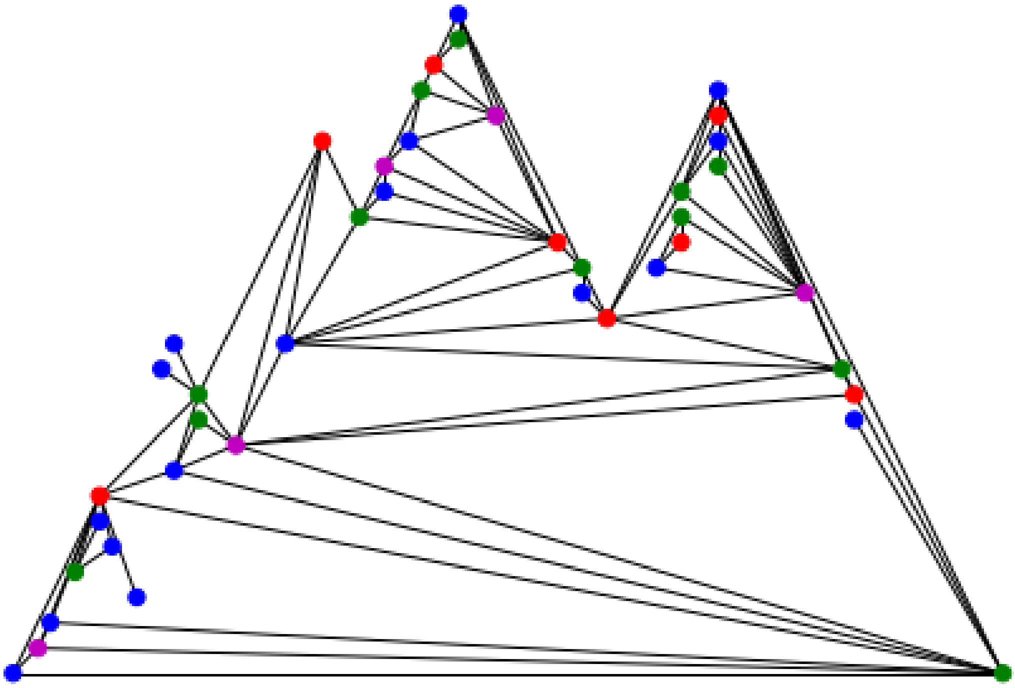
A function $Z : V \to \mathbb{N}$ is a vertex coloring of G *iff* $\forall(v, u) \in E : Z(v) \neq Z(u)$.

A vertex coloring is minimum *iff* $\max_{v \in V} Z(v)$ is minimum

Answer

```
Albania: 2
Andorra: 1
Austria: 4
Belarus: 2
Belgium: 4
Bosnia and Herzegovina: 1
Bulgaria: 1
Croatia: 2
```

```
Czech Republic: 3
Denmark: 1
Estonia: 2
Finland: 2
France: 3
Germany: 2
Greece: 3
Hungary: 1
Italy: 2
Kosovo: 1
Latvia: 1
Liechtenstein: 2
Lithuania: 3
Luxembourg: 1
Moldova: 1
Monaco: 1
Montenegro: 4
Netherlands: 1
North Macedonia: 4
Norway: 1
Poland: 1
Portugal: 1
Romania: 2
Russia: 4
San Marino: 1
Serbia: 3
Slovakia: 2
Slovenia: 3
Spain: 2
Sweden: 3
Switzerland: 1
Turkey: 2
Ukraine: 3
Vatican City: 1
4 colors
```



d

Statement

Find the minimum edge coloring $X : E \to \mathbb{N}$ of G

Definitions

A function $X : E \to \mathbb{N}$ is an edge coloring of G *iff* for any 2 adjacent edges $e_1, e_2 \in E : X(e_1) \neq X(e_2)$

An edge coloring is minimum *iff* $\max\limits_{e \in E} X(e)$ is minimum

Answer

```
Albania - Greece: 0
Albania - Kosovo: 8
Albania - Montenegro: 1
Albania - North Macedonia: 2
Andorra - France: 8
Andorra - Spain: 0
Austria - Czech Republic: 3
Austria - Germany: 4
```

```
Austria - Hungary: 5
Austria - Italy: 0
Austria - Liechtenstein: 8
Austria - Slovakia: 1
Austria - Slovenia: 6
Austria - Switzerland: 2
Belarus - Latvia: 2
Belarus - Lithuania: 1
Belarus - Poland: 5
Belarus - Russia: 3
Belarus - Ukraine: 0
Belgium - France: 1
Belgium - Germany: 3
Belgium - Luxembourg: 8
Belgium - Netherlands: 0
Bosnia and Herzegovina - Croatia: 1
Bosnia and Herzegovina - Montenegro: 0
Bosnia and Herzegovina - Serbia: 2
Bulgaria - Greece: 2
Bulgaria - North Macedonia: 0
Bulgaria - Romania: 3
Bulgaria - Serbia: 5
Bulgaria - Turkey: 1
Croatia - Hungary: 3
Croatia - Montenegro: 2
Croatia - Serbia: 6
Croatia - Slovenia: 0
Czech Republic - Germany: 2
Czech Republic - Poland: 1
Czech Republic - Slovakia: 0
Denmark - Germany: 7
Estonia - Latvia: 1
Estonia - Russia: 4
Finland - Norway: 1
Finland - Russia: 8
Finland - Sweden: 0
France - Germany: 6
France - Italy: 4
France - Luxembourg: 0
France - Monaco: 7
France - Netherlands: 5
France - Spain: 2
France - Switzerland: 3
Germany - Luxembourg: 5
Germany - Netherlands: 1
Germany - Poland: 8
Germany - Switzerland: 0
Greece - North Macedonia: 3
```

```
Greece - Turkey: 8
Hungary - Romania: 8
Hungary - Serbia: 4
Hungary - Slovakia: 7
Hungary - Slovenia: 2
Hungary - Ukraine: 6
Italy - San Marino: 8
Italy - Slovenia: 1
Italy - Switzerland: 5
Italy - Vatican City: 2
Kosovo - Montenegro: 4
Kosovo - North Macedonia: 1
Kosovo - Serbia: 0
Latvia - Lithuania: 0
Latvia - Russia: 5
Liechtenstein - Switzerland: 1
Lithuania - Poland: 4
Lithuania - Russia: 2
Moldova - Romania: 0
Moldova - Ukraine: 8
Montenegro - Serbia: 3
North Macedonia - Serbia: 8
Norway - Russia: 0
Norway - Sweden: 8
Poland - Russia: 6
Poland - Slovakia: 3
Poland - Ukraine: 7
Portugal - Spain: 8
Romania - Serbia: 1
Romania - Ukraine: 4
Russia - Ukraine: 1
Slovakia - Ukraine: 2
9 colors
```

e

Statement

Find the maximum clique $Q \subseteq V$ of G

Definitions

$Q \subseteq V$ is a clique of G *iff* $\forall v, u \in Q : v = u \vee \{v, u\} \in E$

A clique is maximum by cardinality

Answer

```
['Lithuania', 'Belarus', 'Latvia', 'Russia']
4 vertices
```

f

Statement

Find the maximum stable set $S \subseteq V$ of G

Definitions

$S \subseteq V$ is a stable set of G *iff* $\forall v, u \in S : v = u \vee \{v, u\} \notin E$

A stable set is maximum by cardinality

Answer

```
['Moldova', 'Netherlands', 'Monaco', 'Bosnia and Herzegovina', 'Luxembourg',
'Vatican City', 'Sweden', 'Lithuania', 'Andorra', 'Albania', 'Czech Republic',
'Portugal', 'San Marino', 'Denmark', 'Slovenia', 'Estonia', 'Liechtenstein',
'Turkey']
18 vertices
```

g

Statement

Find the maximum matching $M \subseteq E$ of G

Definitions

$M \subseteq E$ is a matching of G *iff* $\forall e_1, e_2 \in M : e_1 = e_2 \vee (e_1 \cap e_2) = \emptyset$

A matching is maximum by cardinality

Answer

```
[('Slovakia', 'Ukraine'), ('Russia', 'Finland'), ('Poland', 'Czech Republic'),
('Lithuania', 'Belarus'), ('Montenegro', 'Kosovo'), ('Spain', 'Portugal'),
('Belgium', 'Luxembourg'), ('Norway', 'Sweden'), ('Andorra', 'France'),
('Albania', 'North Macedonia'), ('Latvia', 'Estonia'), ('Denmark', 'Germany'),
('Romania', 'Bulgaria'), ('Serbia', 'Bosnia and Herzegovina'), ('Croatia',
'Slovenia'), ('Liechtenstein', 'Switzerland'), ('Turkey', 'Greece'), ('Italy',
'San Marino'), ('Austria', 'Hungary')]
19 edges
```

h

Statement

Find the minimum vertex cover $R \subseteq V$ of G

Definitions

$R \subseteq V$ is a vertex cover of G *iff* $\forall e \in E : \exists v \in R : v \in e$

A vertex cover is minimum by cardinality

Answer

```
['Switzerland', 'Slovakia', 'Bulgaria', 'Norway', 'Germany', 'Montenegro',
'Belarus', 'Belgium', 'France', 'Poland', 'Kosovo', 'Italy', 'Latvia', 'Russia',
'Croatia', 'North Macedonia', 'Austria', 'Greece', 'Romania', 'Finland',
'Serbia', 'Spain', 'Ukraine', 'Hungary']
24 vertices
```

i

Statement

Find the minimum edge cover $F \subseteq E$ of G

Definitions

$F \subseteq E$ is an edge cover of G *iff* $\forall v \in V : \exists e \in F : v \in e$

An edge cover is minimum by cardinality

Explanation

We start with a maximum matching and add edges greedily to cover unmatched vertices

A proof of correctness can be found here

Answer

```
[('Slovakia', 'Ukraine'), ('Russia', 'Finland'), ('Poland', 'Czech Republic'),
('Lithuania', 'Belarus'), ('Montenegro', 'Kosovo'), ('Spain', 'Portugal'),
('Belgium', 'Luxembourg'), ('Norway', 'Sweden'), ('Andorra', 'France'),
('Albania', 'North Macedonia'), ('Latvia', 'Estonia'), ('Denmark', 'Germany'),
('Romania', 'Bulgaria'), ('Serbia', 'Bosnia and Herzegovina'), ('Croatia',
'Slovenia'), ('Liechtenstein', 'Switzerland'), ('Turkey', 'Greece'), ('Italy',
'San Marino'), ('Austria', 'Hungary'), ('Belgium', 'Netherlands'), ('France',
'Monaco'), ('Italy', 'Vatican City'), ('Moldova', 'Romania')]
23 edges
```

j

Statement

Find the shortest closed walk W that visits every vertex of G

Definitions

A hamiltonian cycle of G is a cycle that contains all vertices.

The traveling salesman problem is the problem of finding the hamiltonian cycle of G of minimum length (in the case of a weigted graph length is defined as the total weight of edges).

The distance graph of G is a complete graph with the same vertice set, in which each edge has a weight equal to the distance between its endpoints in G.

Explanation

The problem can be reduced to the traveling salesman problem on the distance graph of G.

After obtaining the minimum hamiltonian cycle of the distance graph, we replace every edge in it with the shortest path between its endpoints in G.

Here we use an approximate algorithm because the traveling salesman problem is apparently already too hard for my pc even with this relatively small graph. the commented cell contains the exact evaluation that didn't finish in an hour for me

Answer

```
['Austria', 'Liechtenstein', 'Switzerland', 'Germany', 'Denmark', 'Germany',
'Luxembourg', 'Belgium', 'Netherlands', 'France', 'Monaco', 'France', 'Andorra',
'Spain', 'Portugal', 'Spain', 'France', 'Italy', 'San Marino', 'Italy', 'Vatican
City', 'Italy', 'Slovenia', 'Croatia', 'Serbia', 'Bosnia and Herzegovina',
'Montenegro', 'Kosovo', 'North Macedonia', 'Albania', 'Greece', 'Turkey',
'Bulgaria', 'Romania', 'Moldova', 'Ukraine', 'Belarus', 'Lithuania', 'Latvia',
'Estonia', 'Russia', 'Finland', 'Sweden', 'Norway', 'Russia', 'Poland', 'Czech
Republic', 'Slovakia', 'Hungary', 'Austria']
49 edges
```

k

Statement

Find the shortest closed walk U that visits every edge of G

Definitions

A eulerian circuit is a ciruit that visits each edge exactly once

Explanation

In order solve this problem we can add parallel edges to the graph and then find a eulerian curcuit in the produced multigraph. Essentially we're reframing our question by saying that visiting the same edge several times is the same as visiting each of several parallel edges exactly once. The problem then becomes that of adding optimal parallel edges, which thankfully networkx is capable of.

Answer

```
['Moldova', 'Ukraine', 'Romania', 'Ukraine', 'Russia', 'Latvia', 'Estonia',
'Russia', 'Poland', 'Ukraine', 'Hungary', 'Serbia', 'Romania', 'Hungary',
'Croatia', 'Hungary', 'Austria', 'Slovenia', 'Hungary', 'Slovakia', 'Ukraine',
'Belarus', 'Poland', 'Belarus', 'Russia', 'Finland', 'Sweden', 'Norway',
'Finland', 'Norway', 'Russia', 'Lithuania', 'Belarus', 'Latvia', 'Lithuania',
'Poland', 'Czech Republic', 'Austria', 'Germany', 'Denmark', 'Germany',
'Poland', 'Slovakia', 'Austria', 'Italy', 'San Marino', 'Italy', 'Vatican City',
'Italy', 'France', 'Spain', 'Portugal', 'Spain', 'Andorra', 'France', 'Belgium',
'Germany', 'France', 'Monaco', 'France', 'Luxembourg', 'Germany', 'Netherlands',
'France', 'Luxembourg', 'Belgium', 'Netherlands', 'France', 'Switzerland',
'Germany', 'Czech Republic', 'Slovakia', 'Austria', 'Switzerland', 'Austria',
'Liechtenstein', 'Switzerland', 'Italy', 'Slovenia', 'Croatia', 'Serbia', 'North
Macedonia', 'Greece', 'Albania', 'Kosovo', 'Serbia', 'Montenegro', 'Kosovo',
'North Macedonia', 'Albania', 'Montenegro', 'Croatia', 'Bosnia and Herzegovina',
'Montenegro', 'Bosnia and Herzegovina', 'Serbia', 'Bulgaria', 'Greece',
'Turkey', 'Bulgaria', 'North Macedonia', 'Bulgaria', 'Romania', 'Moldova']
103 edges
```

l

Find all 2-vertex-connected components (blocks) and draw a block-cut tree of G^*

Definitions

A cut vertex is a vertex the removal of which splits the graph, increasing its number of connected components

A graph is 2-vertex-connected *iff* it contains no cut vertices.

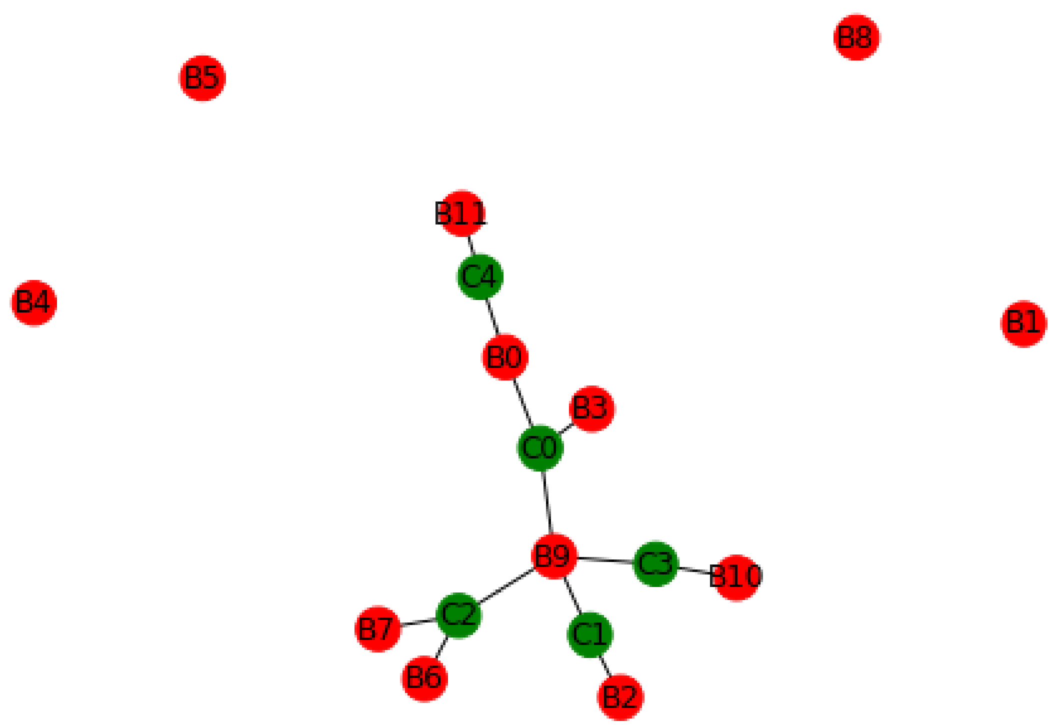
A block of G is a 2-vertex-connected subgraph of G that is maximal by inclusion.

A block-cut tree of G is a tree which has a vertex for each block and each cut vertex of G and a edge connecting each pair of vertices corresponding to a cut vertex and a block for which the vertex is in the block

Answer

```
B0: ('Andorra', 'France', 'Spain')
B1: ('Cyprus',)
B2: ('Denmark', 'Germany')
```

```
B3: ('France', 'Monaco')
B4: ('Iceland',)
B5: ('Ireland', 'United Kingdom')
B6: ('Italy', 'San Marino')
B7: ('Italy', 'Vatican City')
B8: ('Malta',)
B9: ('Moldova', 'Romania', 'Ukraine', 'Liechtenstein', 'Switzerland', 'Austria',
'France', 'Germany', 'Italy', 'Belgium', 'Netherlands', 'Bosnia and
Herzegovina', 'Montenegro', 'Croatia', 'Serbia', 'Czech Republic', 'Slovakia',
'Poland', 'Hungary', 'Bulgaria', 'North Macedonia', 'Turkey', 'Greece',
'Luxembourg', 'Latvia', 'Lithuania', 'Belarus', 'Russia', 'Albania', 'Kosovo',
'Slovenia', 'Estonia')
B10: ('Norway', 'Russia', 'Finland', 'Sweden')
B11: ('Portugal', 'Spain')
C0: France
C1: Germany
C2: Italy
C3: Russia
C4: Spain
```



m

Statement

Find all 2-edge-connected components of G^*

Definitions

A bridge is an edge the removal of which splits the graph, increasing its number of connected components

A graph is 2-edge-connected *iff* it contains no bridges

A 2-edge-connected component of G is a 2-edge-connected subgraph of G that is maximal by inclusion

Answer

```
['United Kingdom']
['Moldova', 'Switzerland', 'Netherlands', 'Slovakia', 'Bosnia and Herzegovina',
'Bulgaria', 'Norway', 'Luxembourg', 'Germany', 'Sweden', 'Lithuania', 'Andorra',
'Montenegro', 'Belarus', 'Czech Republic', 'Albania', 'Belgium', 'France',
'Poland', 'Kosovo', 'Italy', 'Slovenia', 'Estonia', 'Latvia', 'Russia',
'Croatia', 'Turkey', 'North Macedonia', 'Liechtenstein', 'Austria', 'Serbia',
'Romania', 'Finland', 'Greece', 'Spain', 'Ukraine', 'Hungary']
['Monaco']
['Vatican City']
['Malta']
['Ireland']
['Portugal']
['Cyprus']
['San Marino']
['Denmark']
['Iceland']
```

n

Statement

Construct an SPQR tree of the largest biconnected component of G

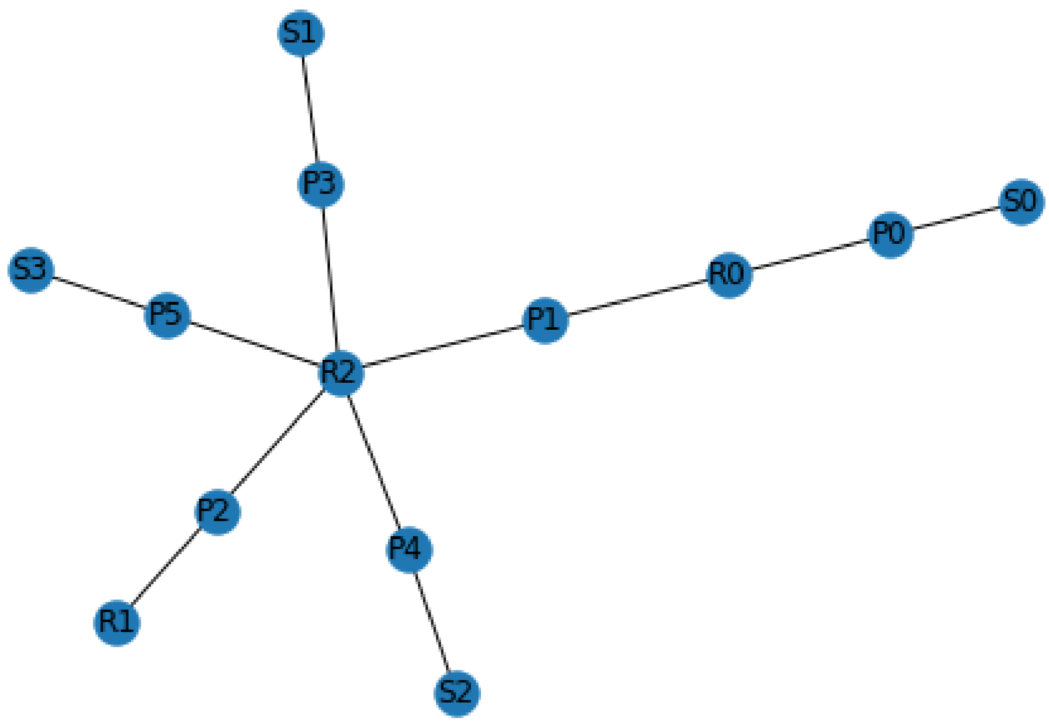
Definitions

An SPQR tree is a tree representing the triconnected components of a biconnected graph, the exact structure of which is too long to explain here and can be found on Wikipedia

Answer

```
S0: ['Estonia', 'Latvia', 'Russia']
S1: ['Austria', 'Liechtenstein', 'Switzerland']
```

```
S2: ['Bulgaria', 'Greece', 'Turkey']
S3: ['Moldova', 'Romania', 'Ukraine']
P0: ['Latvia', 'Russia']
P1: ['Poland', 'Ukraine']
P2: ['France', 'Germany']
P3: ['Austria', 'Switzerland']
P4: ['Bulgaria', 'Greece']
P5: ['Romania', 'Ukraine']
R0: ['Belarus', 'Latvia', 'Lithuania', 'Poland', 'Russia', 'Ukraine']
R1: ['Belgium', 'France', 'Germany', 'Luxembourg', 'Netherlands']
R2: ['Albania', 'Austria', 'Bosnia and Herzegovina', 'Bulgaria', 'Croatia',
'Czech Republic', 'France', 'Germany', 'Greece', 'Hungary', 'Italy', 'Kosovo',
'Montenegro', 'North Macedonia', 'Poland', 'Romania', 'Serbia', 'Slovakia',
'Slovenia', 'Switzerland', 'Ukraine']
```



O

Statement

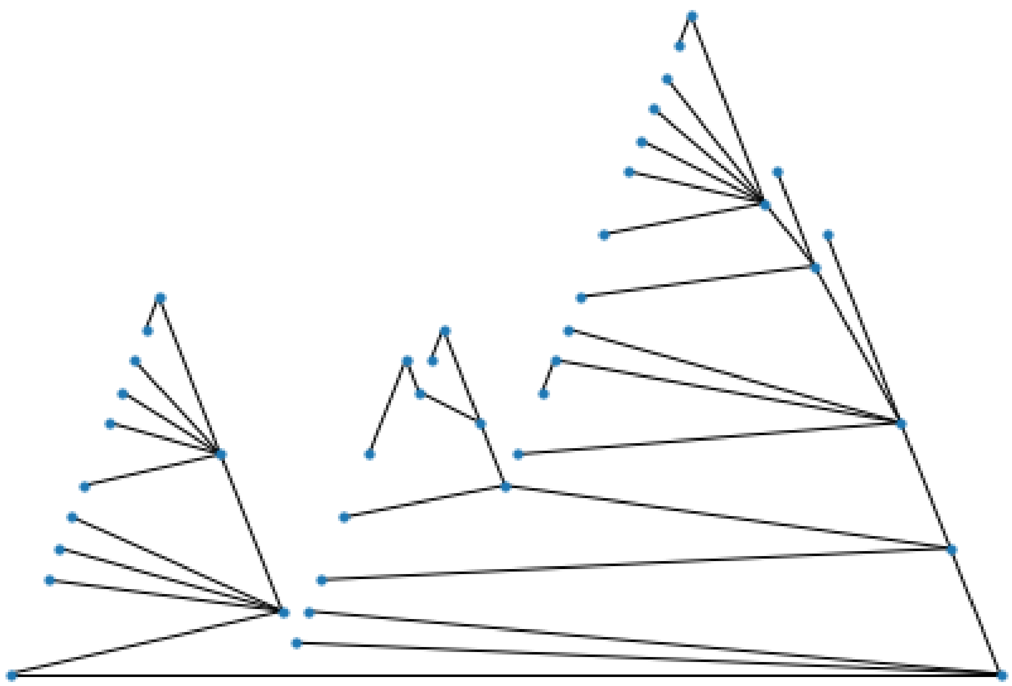
Find the minimum (w.r.t. the total weight of edges) spanning tree T for the maximum connected component of the weighted version of G

Definition

A spanning tree of G is a tree with the same vertex set as G

Answer

`22164.45282255872`



p

Statement

Find the $centroid(T)$ (w.r.t. the edge weight function).

Definitions

A branch of a tree T starting at one of its vertices v is a subtree of T in which v has exactly one neighbor.

The centroid of a tree (w.r.t. edge weight) is the set of vertices with minimum weight, where we define the weight of a vertex to be the maximum total edge weight of a branch starting at that vertex

Answer

```
['Hungary']
```

q

Statement

Construct the Prüfer code for T

Definitions

The Prüfer code of a tree labeled with integers $\{1, 2, ..., n\}$ is a sequence of $n-2$ labels that uniquely identifies that tree.

Explanation

We sort the vertices lexicographically to contstuct the code.

Answer

```
[12, 40, 12, 7, 30, 2, 13, 31, 2, 31, 2, 31, 12, 12, 12, 17, 24, 31, 40, 36, 16,
30, 40, 15, 12, 16, 11, 31, 40, 2, 14, 0, 24, 7, 15, 22, 30, 15, 2, 16]
```