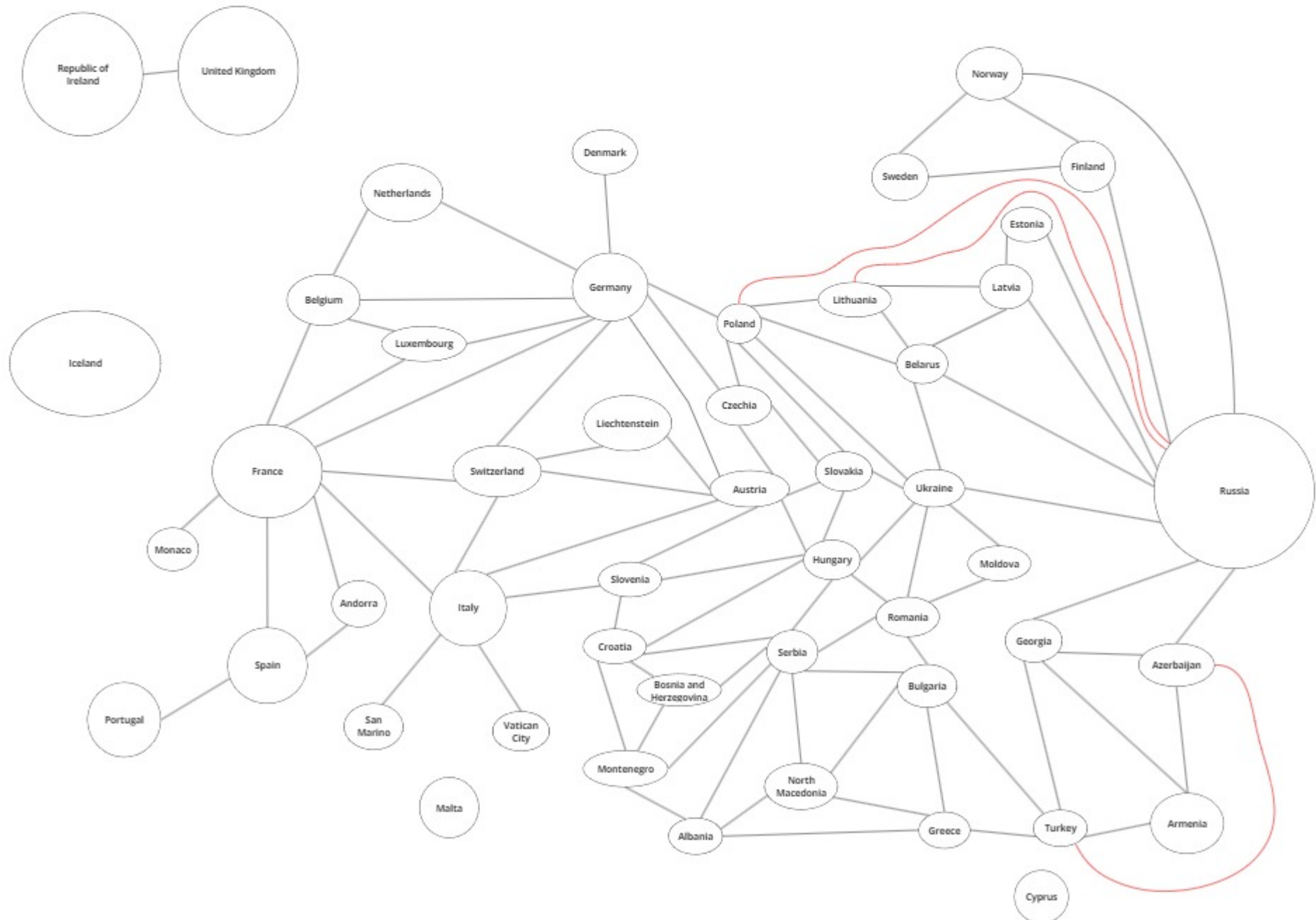


1. Постройте планарный граф G. Если это невозможно, докажите почему (hint: критерий Понтрягина-Куратовского)

Изобразим граф так, чтобы никакие ребра не пересекались между собой (ребра, которые изобразили иначе по сравнению с исходным графом выделены **красным** цветом)

Изобразить ребра таким способом (чтобы они не пересекались) получилось, следовательно, текущий граф **является планарным**



2. Найдите $|V|$, $|E|$, $\delta(G) = \min(\deg(v)) \ v \in V$, $\Delta(G) = \max(\deg(v)) \ v \in V$, радиус, диаметр, центр, цикломатическое число

Прикрепляю файл Excel, где есть таблица с путями. Ответы выделены зеленым цветом справа от таблицы, а также изображены снизу



Количество вершин ($ V $)	Количество ребер ($ E $)	Минимальная степень вершины ($\delta(G) = \min(\deg(v)) \ v \in V$)	Максимальная степень вершины ($\Delta(G) = \max(\deg(v)) \ v \in V$)
49	93	0 (Malta, Iceland, Cyprus)	10 (Russia)

Радиус	Диаметр	Центр	Цикломатическое число
4	8	Poland	49, если считать просто по формуле

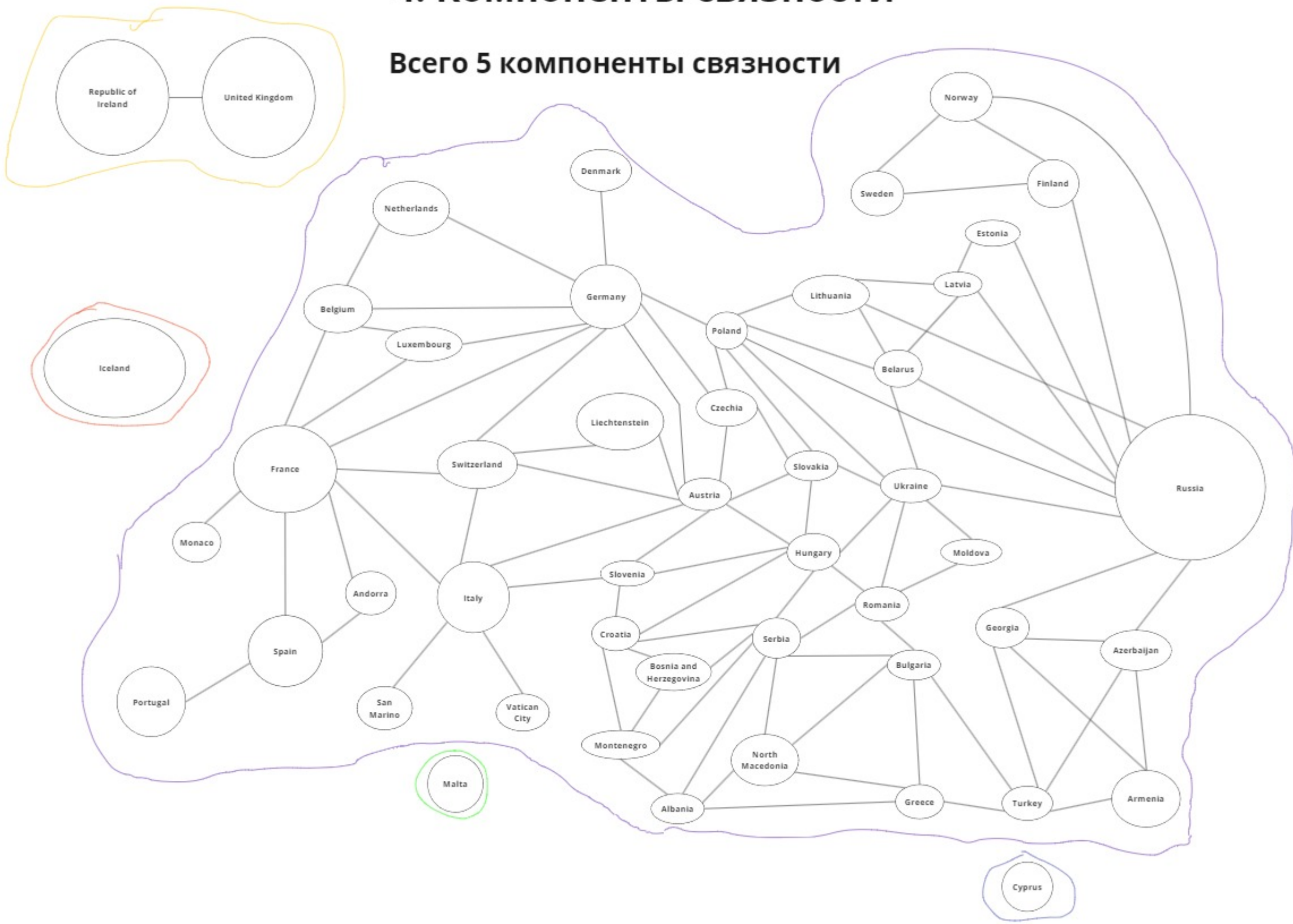
3. Раскраска графа

Хроматическое число: $\chi(G) = 4$



4. Компоненты связности

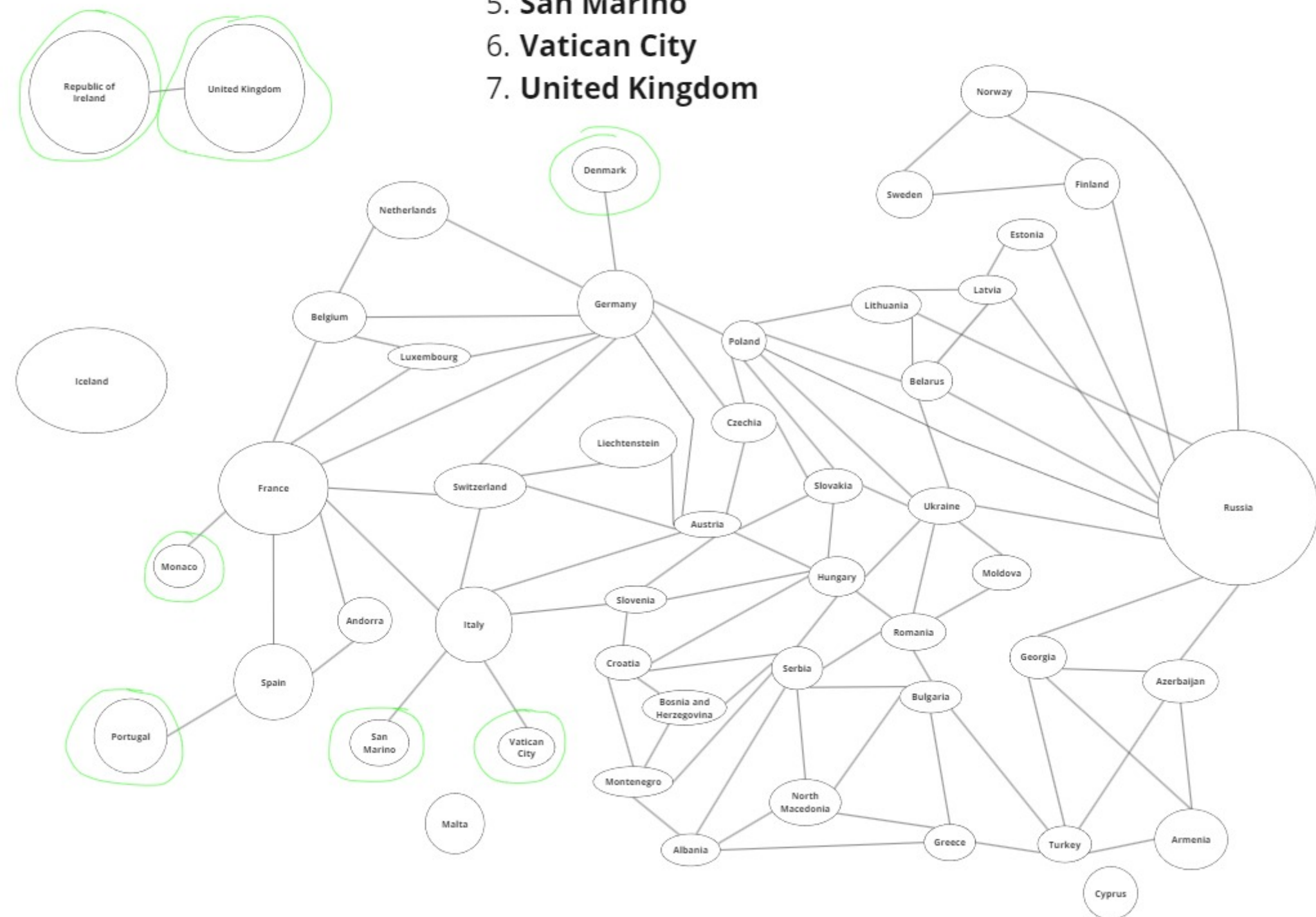
Всего 5 компоненты связности



5. Висячие вершины графа

Всего 7 висячих вершин:

1. Republic of Ireland
2. Denmark
3. Monaco
4. Portugal
5. San Marino
6. Vatican City
7. United Kingdom



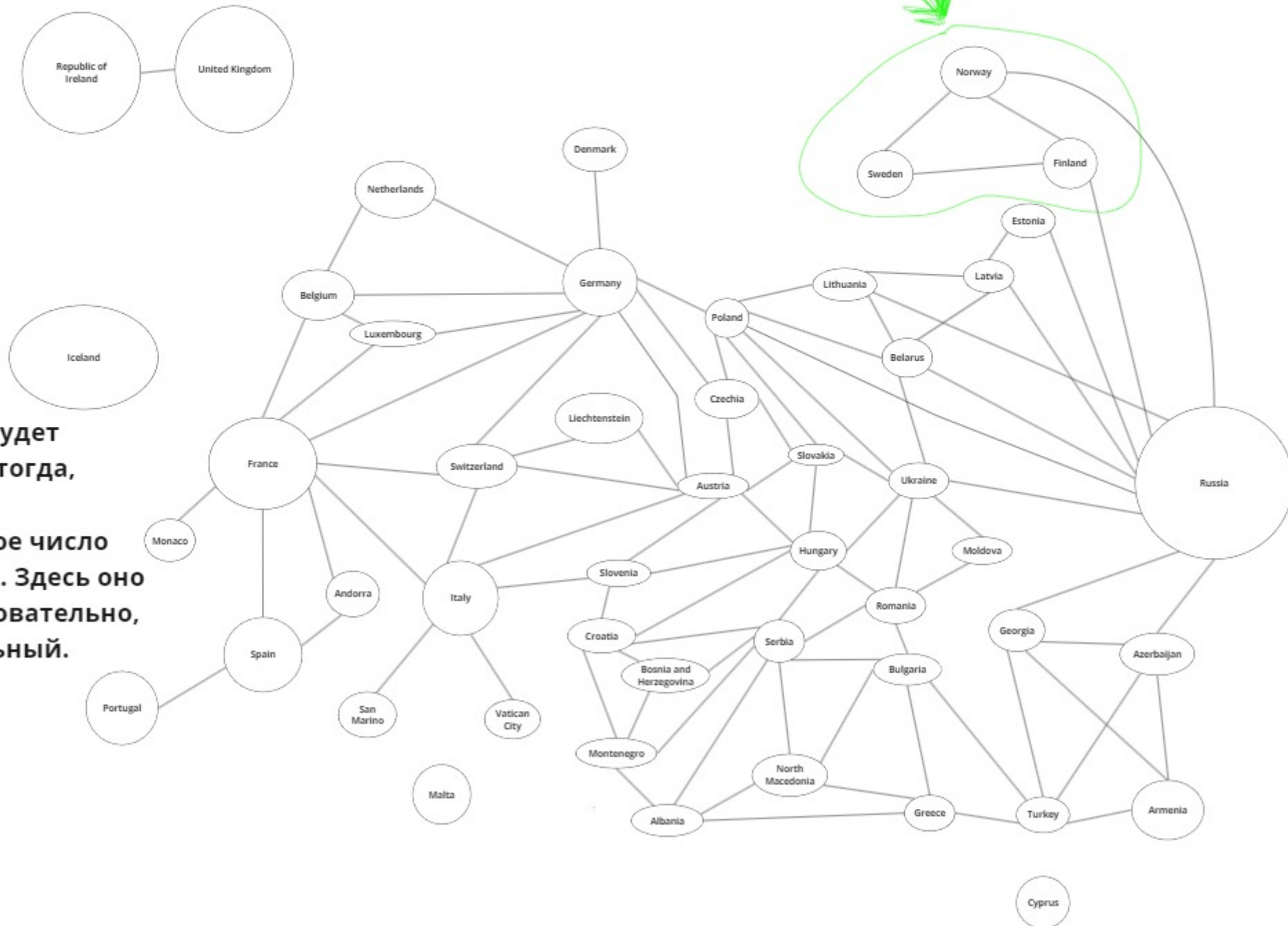
6. Проверка на двудольность

"Граф является двудольным тогда и только тогда, когда все (простые) циклы в графе имеют чётную длину."

Обратим внимание, что есть множество циклов нечетной длины, например:

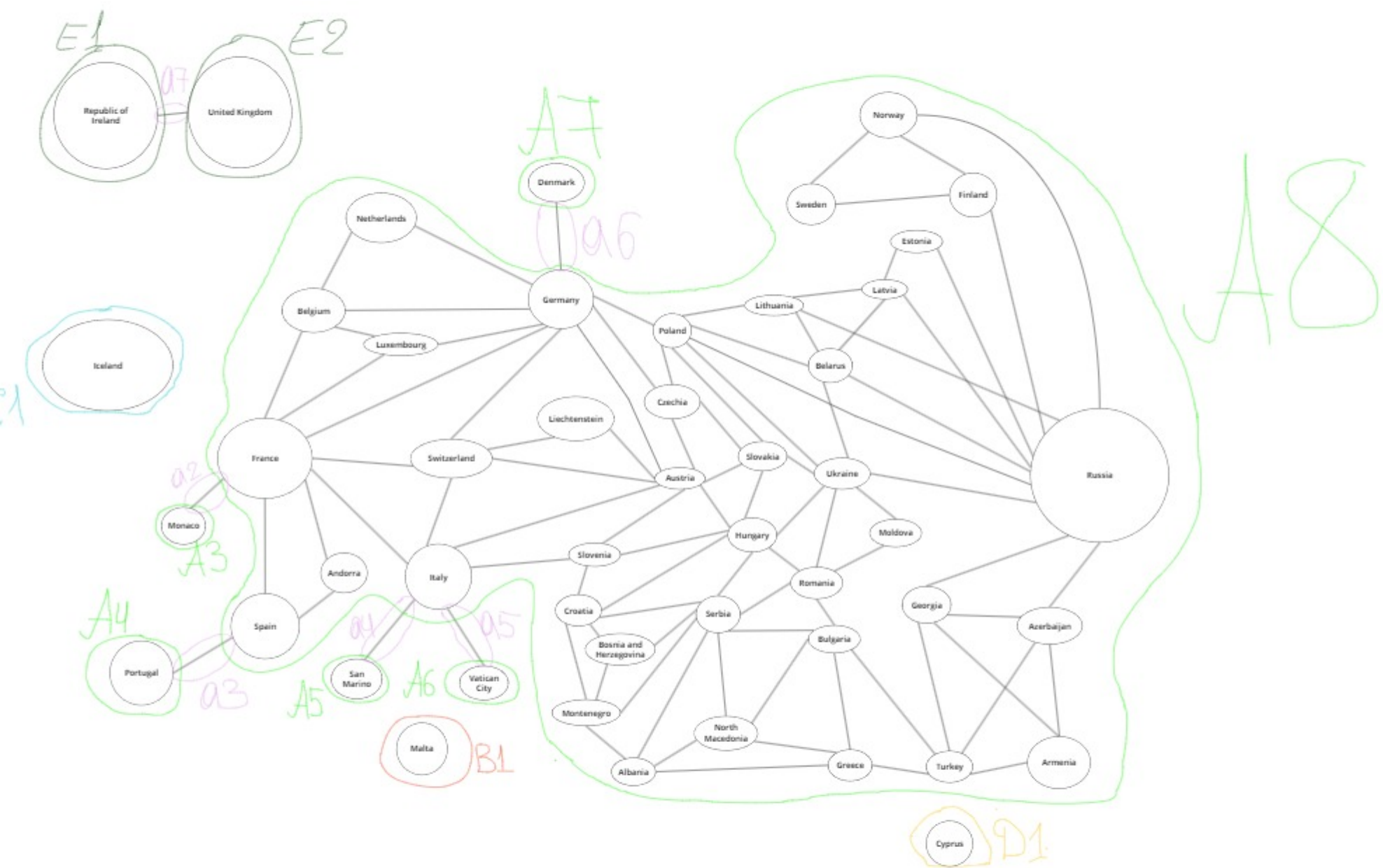
Norway - Sweden - Finland - Norway.

Следовательно, граф не является двудольным.



Также, граф будет двудольным тогда, когда его хроматическое число будет равно 2. Здесь оно равно 4. следовательно, он не двудольный.

7. Найти компоненты реберной двусвязности и построить граф компонент реберной двусвязности



Граф компонент реберной двусвязности

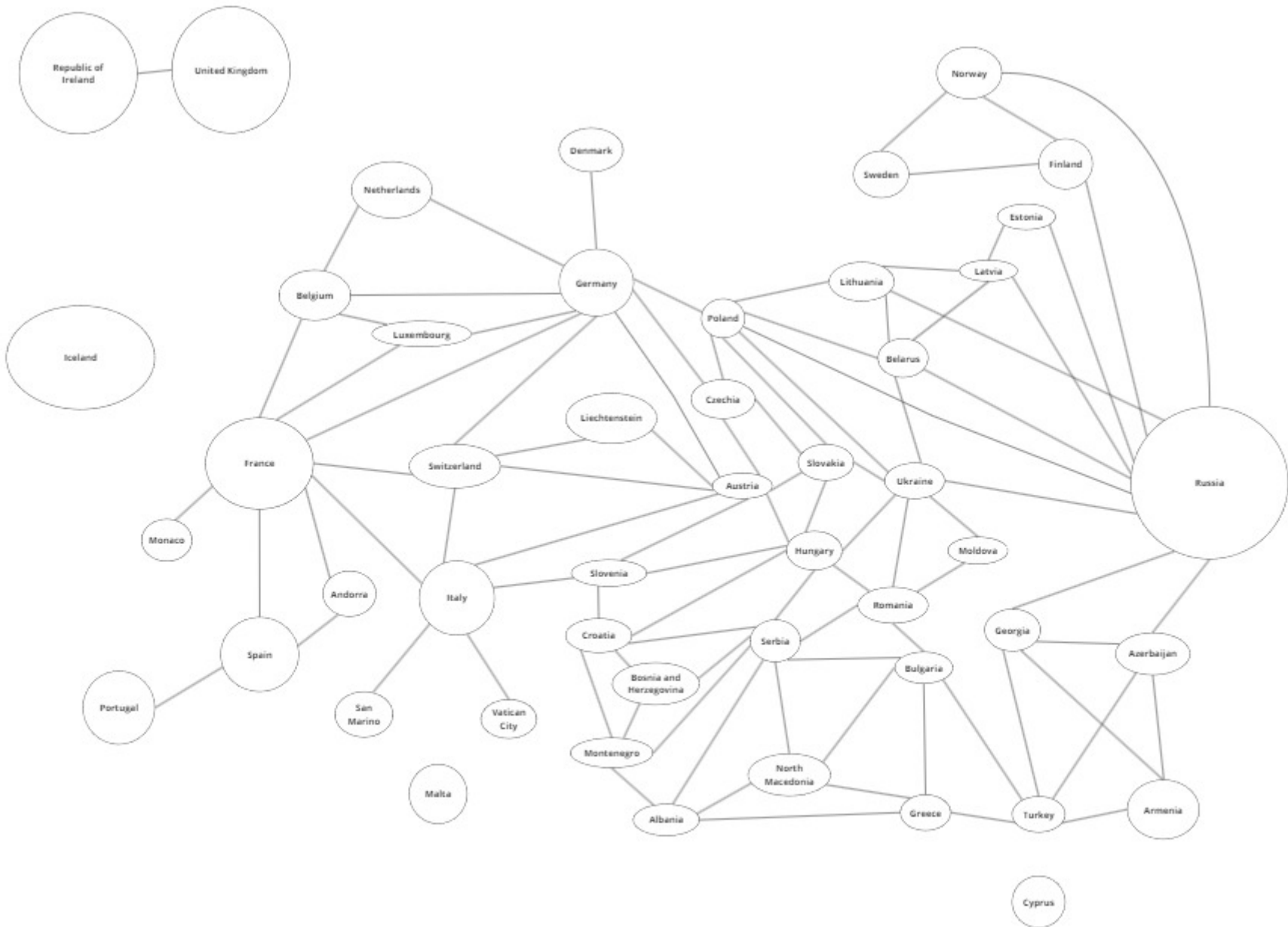
Получается лес, состоящий из 3 деревьев.



9. Найти кратчайший замкнутый путь W , который проходит по всем вершинам G

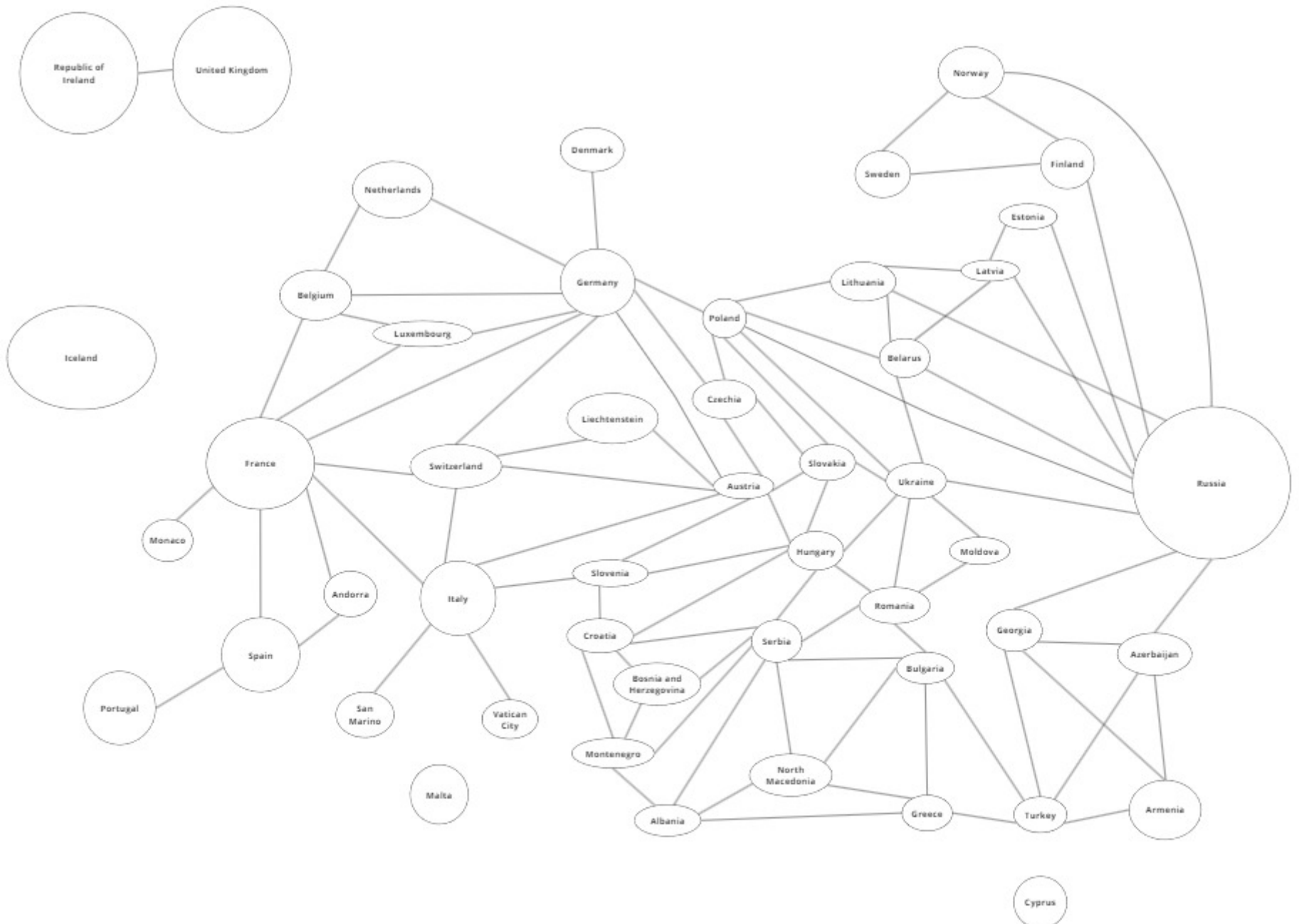
Ответ: так как надо найти путь, который будет проходить по всем вершинам по одному разу, а также нужно, чтобы путь был замкнутым, то требуется найти гамильтонов цикл.

Рассмотрим любую из висячих вершин, например, Данию, она связана с Германией. Чтобы посетить Данию придется зайти в Германию, потом зайти в Данию, а потом обратно в Германию. Из этого идет противоречие, так как мы посетим Данию только в том случае, если два раза зайдем в Германию. Так будет работать со всеми висячими вершинами. Следовательно, такой путь найти невозможно.



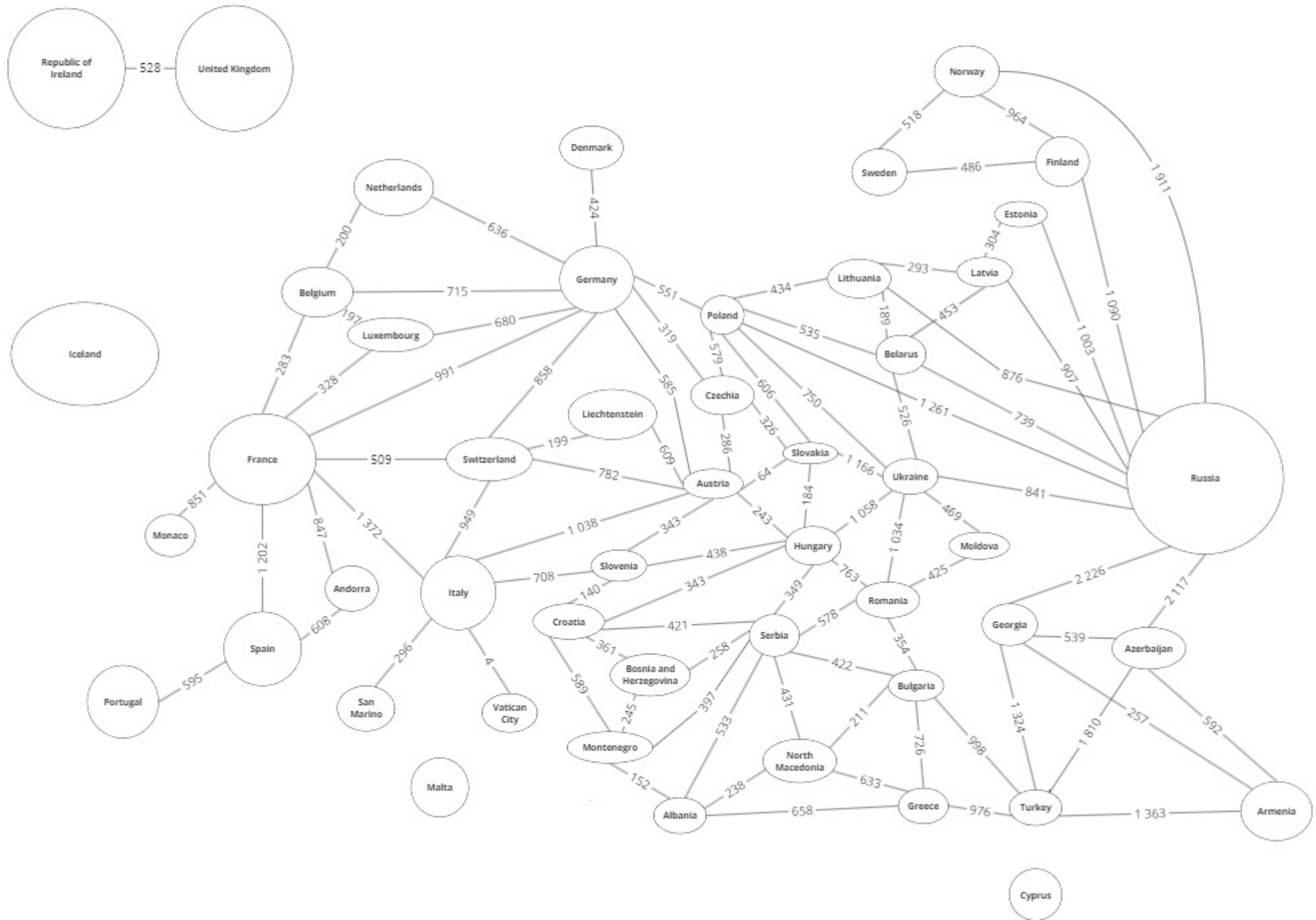
10. Найти кратчайший замкнутый путь U , который проходит по всем ребрам G

Ответ: так как надо найти путь, который будет проходить по всем ребрам по одному разу, а также нужно, чтобы путь был замкнутым, то требуется найти эйлеров цикл. Рассмотрим любое из висячих ребер, например, ребро (Дания, Германия). Чтобы посетить это ребро, нужно будет зайти в Данию, а потом обратно в Германию. Из этого идет противоречие, так как мы посетим ребро (Дания, Германия) два раза. Так будет работать со всеми висячими ребрами. Следовательно, такой путь найти невозможно.



11. Добавить веса на ребра, равные расстоянию между столицами двух стран.
Найти минимальное остовное дерево T для максимальной компоненты связности взвешенного графа G

Граф с весами



Примечание:

Расстояния между столицами брались с помощью google maps.

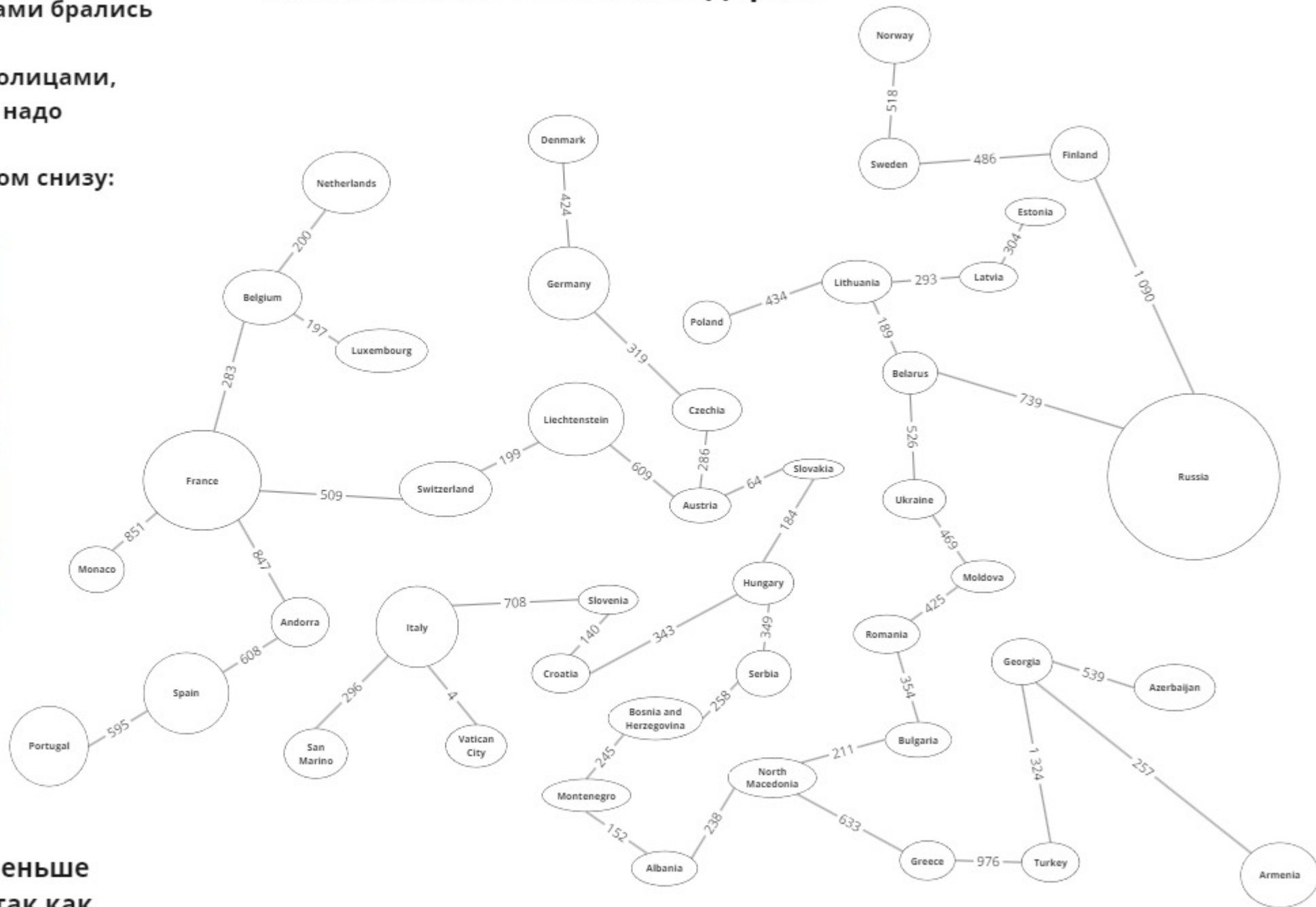
Брал расстояния между столицами, учитывая, что добираться надо пешком.

Пример м/у Римом и Бёрном снизу:



222 ч (949 км)
Через SS 9 Via Emilia

Минимальное остовное дерево



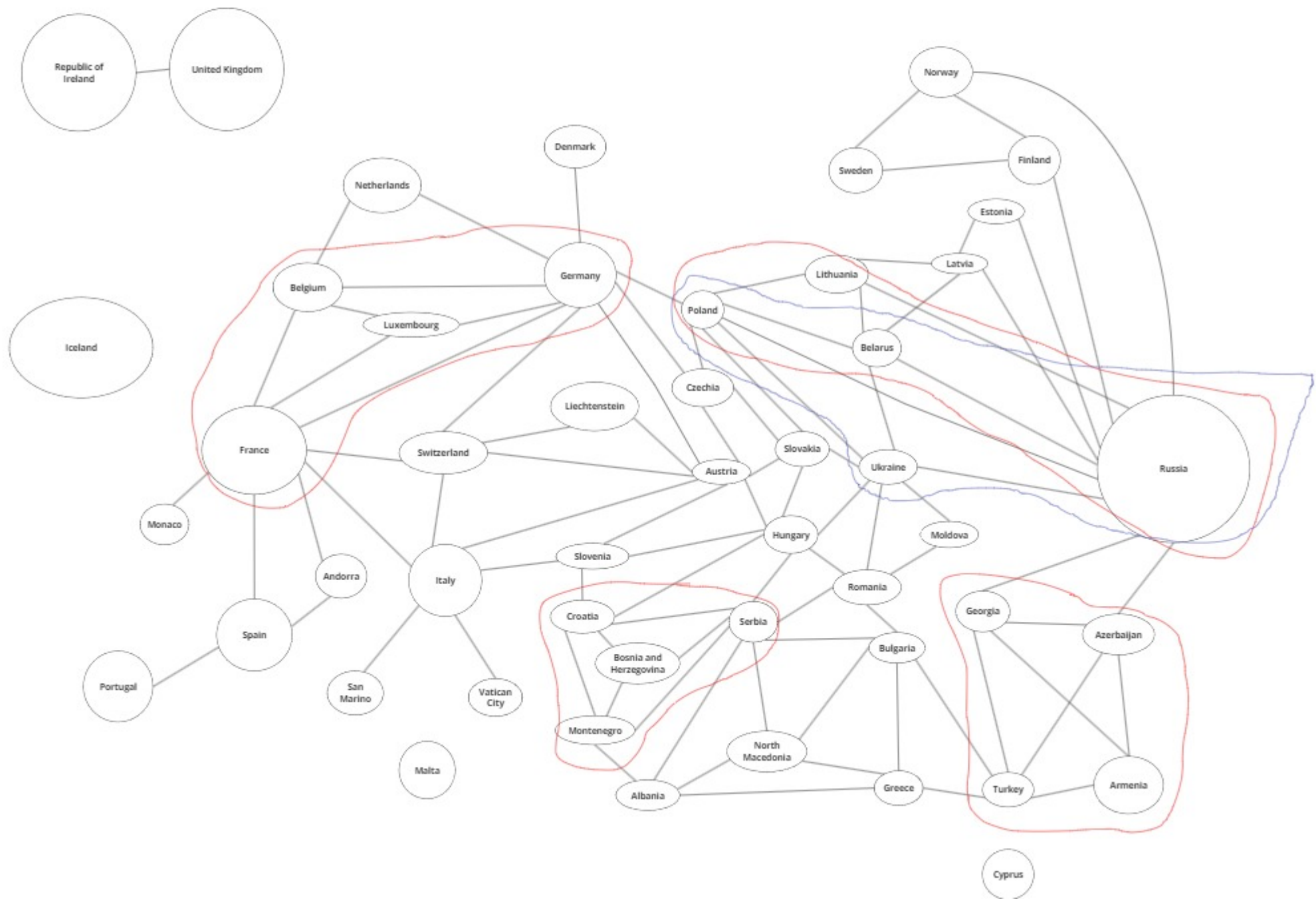
Примечание:

Получается число на 1 меньше (43) цикломатического, так как мы не считаем здесь ребро между Великобританией и Ирландией.

12. Клика – полный подграф в графе. Указать максимальную клику этого графа

Заметим, что хроматическое число (у нас оно равно 4) равно размеру наибольшей клики в графе, следовательно, максимальная клика графа равна 4.

Отметим максимальные клики в графе:



13. Построить код Прюфера для графа T



Ответ (брал в алфавитном порядке): Georgia, Germany, Latvia, Czechia, Austria, Turkey, Lithuania, Belgium, France, Sweden, Hungary, Greece, North Macedonia, Moldova, Romania, Bulgaria, North Macedonia, Albania, Montenegro, Bosnia and Herzegovina, Serbia, Hungary, Croatia, Slovenia, Italy.

~~Albania~~, ~~Andorra~~, ~~Armenia~~, ~~Austria~~, ~~Azerbaijan~~, ~~Belarus~~, ~~Belgium~~, ~~BosniaAndHerzegovina~~, ~~Bulgaria~~, ~~Croatia~~, ~~CzechRepublic~~, ~~Denmark~~, ~~Estonia~~, ~~Finland~~, ~~France~~, ~~Germany~~, ~~Georgia~~, ~~Greece~~, ~~Hungary~~, Italy, ~~Latvia~~, ~~Liechtenstein~~, ~~Lithuania~~, ~~Luxembourg~~, ~~Moldova~~, ~~Monaco~~, ~~Montenegro~~, ~~Netherlands~~, ~~RepublicOfMacedonia~~, ~~Norway~~, ~~Poland~~, ~~Portugal~~, ~~Romania~~, ~~Russia~~, ~~SanMarino~~, ~~Serbia~~, ~~Slovakia~~, ~~Slovenia~~, ~~Spain~~, ~~Sweden~~, ~~Switzerland~~, ~~Turkey~~, ~~Ukraine~~, VaticanCity