Guion 4:

Actividad 1:

Salida del programa:

-------- Lista de Países --------
Malta: (Rojo)
Chipre: (Rojo)
Albania: (Rojo)
Montenegro (Amarillo)
Macedonia (Azul)
Grecia (Verde)
Ciudad del Vaticano: (Rojo)
Italia (Azul)
Ucrania: (Rojo)
Rusia (Amarillo)
Bielorrusia (Azul)
Polonia (Negro)
Eslovaquia (Azul)
Hungría (Verde)
Rumania (Amarillo)
Moldavia (Azul)
Lituania: (Rojo)
Letonia (Verde)
Polonia (Negro)
Rusia (Amarillo)
Bielorrusia (Azul)
Luxemburgo: (Rojo)
Francia (Amarillo)
Alemania (Verde)
Bélgica (Azul)
Georgia: (Rojo)
Rusia (Amarillo)
Turquía (Azul)
Armenia (Amarillo)
Azerbaiyán (Verde)
San Marino: (Rojo)
Italia (Azul)
República Checa: (Rojo)
Polonia (Negro)
Eslovaquia (Azul)
Austria (Amarillo)
Alemania (Verde)
Turquía: (Azul)
Grecia (Verde)
Bulgaria (Negro)
Georgia (Rojo)
Armenia (Amarillo)
Azerbaiyán (Verde)
Bielorrusia: (Azul)
Lituania (Rojo)
Letonia (Verde)
Rusia (Amarillo)
Ucrania (Rojo)
Polonia (Negro)
Irlanda: (Rojo)
Reino Unido (Azul)
Eslovaquia: (Azul)
Polonia (Negro)
República Checa (Rojo)
Austria (Amarillo)
Hungría (Verde)
Ucrania (Rojo)
Dinamarca: (Rojo)
Alemania (Verde)
Italia: (Azul)
Francia (Amarillo)
Suiza (Negro)
Austria (Amarillo)
Eslovenia (Rojo)
Ciudad del Vaticano (Rojo)
San Marino (Rojo)
Islandia: (Rojo)
Serbia: (Rojo)
Hungría (Verde)
Rumania (Amarillo)
Bulgaria (Negro)
Macedonia (Azul)
Bosnia y Herzegovina (Verde)
Croacia (Azul)
Montenegro (Amarillo)
Bélgica: (Azul)
Países Bajos (Rojo)
Alemania (Verde)
Luxemburgo (Rojo)
Francia (Amarillo)
Alemania: (Verde)
Francia (Amarillo)
Suiza (Negro)
Austria (Amarillo)
Bélgica (Azul)
Países Bajos (Rojo)
Luxemburgo (Rojo)
Dinamarca (Rojo)
Polonia (Negro)
República Checa (Rojo)
Hungría: (Verde)
Eslovaquia (Azul)
Ucrania (Rojo)
Rumania (Amarillo)
Serbia (Rojo)
Croacia (Azul)
Eslovenia (Rojo)
Austria (Amarillo)
Andorra: (Rojo)
España (Azul)
Francia (Amarillo)
Moldavia: (Azul)
Rumania (Amarillo)
Ucrania (Rojo)
Croacia: (Azul)
Bosnia y Herzegovina (Verde)
Montenegro (Amarillo)
Eslovenia (Rojo)
Hungría (Verde)
Serbia (Rojo)
Portugal: (Rojo)
España (Azul)
Azerbaiyán: (Verde)
Georgia (Rojo)
Rusia (Amarillo)
Armenia (Amarillo)
Turquía (Azul)
Noruega: (Rojo)
Rusia (Amarillo)
Finlandia (Verde)
Suecia (Azul)
Liechtenstein: (Rojo)
Suiza (Negro)
Austria (Amarillo)
Bosnia y Herzegovina: (Verde)
Croacia (Azul)
Serbia (Rojo)
Montenegro (Amarillo)
Rusia: (Amarillo)
Noruega (Rojo)
Finlandia (Verde)
Estonia (Rojo)
Letonia (Verde)
Bielorrusia (Azul)
Lituania (Rojo)
Polonia (Negro)
Ucrania (Rojo)
Georgia (Rojo)
Azerbaiyán (Verde)
Kazajistán (Rojo)
Países Bajos: (Rojo)
Bélgica (Azul)
Alemania (Verde)
Grecia: (Verde)
Albania (Rojo)
Bulgaria (Negro)
Macedonia (Azul)
Turquía (Azul)

Suecia: (Azul)
- Noruega (Rojo)
- Finlandia (Verde)

Reino Unido: (Azul)
- Irlanda (Rojo)

Macedonia: (Azul)
- Serbia (Rojo)
- Montenegro (Amarillo)
- Bulgaria (Negro)
- Grecia (Verde)
- Albania (Rojo)

Francia: (Amarillo)
- España (Azul)
- Italia (Azul)
- Suiza (Negro)
- Alemania (Verde)
- Luxemburgo (Rojo)
- Bélgica (Azul)
- Andorra (Rojo)
- Mónaco (Rojo)

Rumania: (Amarillo)
- Ucrania (Rojo)
- Moldavia (Azul)
- Ucrania (Rojo)
- Bulgaria (Negro)
- Serbia (Rojo)
- Hungría (Verde)

Polonia: (Negro)
- Rusia (Amarillo)
- Lituania (Rojo)
- Bielorrusia (Azul)
- Ucrania (Rojo)
- República Checa (Rojo)
- Eslovaquia (Azul)
- Alemania (Verde)

Austria: (Amarillo)
- Alemania (Verde)
- República Checa (Rojo)
- Eslovaquia (Azul)
- Hungría (Verde)
- Eslovenia (Rojo)
- Italia (Azul)
- Suiza (Negro)
- Liechtenstein (Rojo)

Kazajistán: (Rojo)
- Rusia (Amarillo)

Finlandia: (Verde)
- Rusia (Amarillo)
- Suecia (Azul)
- Noruega (Rojo)

Montenegro: (Amarillo)
- Croacia (Azul)
- Bosnia y Herzegovina (Verde)
- Albania (Rojo)
- Serbia (Rojo)
- Macedonia (Azul)

Armenia: (Amarillo)
- Georgia (Rojo)
- Turquía (Azul)
- Azerbaiyán (Verde)

España: (Azul)
- Andorra (Rojo)
- Francia (Amarillo)
- Portugal (Rojo)

Mónaco: (Rojo)
- Francia (Amarillo)

Suiza: (Negro)
- Francia (Amarillo)
- Alemania (Verde)
- Austria (Amarillo)
- Liechtenstein (Rojo)
- Italia (Azul)

Letonia: (Verde)
- Estonia (Rojo)
- Rusia (Amarillo)
- Lituania (Rojo)
- Bielorrusia (Azul)

Estonia: (Rojo)
- Letonia (Verde)
- Rusia (Amarillo)

Eslovenia: (Rojo)
- Austria (Amarillo)
- Hungría (Verde)
- Croacia (Azul)
- Italia (Azul)

Bulgaria: (Negro)
- Rumania (Amarillo)
- Turquía (Azul)
- Grecia (Verde)
- Macedonia (Azul)
- Serbia (Rojo)
- Macedonia (Azul)

El número cromático es: 5

Se han usado los siguientes colores:
1. Rojo
2. Azul
3. Verde
4. Amarillo
5. Negro

Respuesta a las cuestiones:

A) Explica brevemente cómo funciona tu algoritmo.

*Al crear una instancia de la clase **ColoracionMapa.java**, se cargan automáticamente los ficheros de colores y fronteras (se ha tomado esta decisión para realizar más eficiente la creación de dicha instancia).*

*Una vez hecho esto, se llama al método **updateReferences()**, el cual se encarga de que la lista de fronteras contenga los mismos objetos **Country** que los "key" del countriesMap, para ello, se recorre la lista de países, se recorre la lista de fronteras de cada uno de ellos, y se vuelve a recorrer la lista de países, cuando en una iteración el país concuerde con el de la frontera, esta se sustituirá por ese objeto país, de esta forma se consigue que al modificar el color de un país, se modifique en todos los lugares donde aparezca.*

*Una vez hecho esto, se llama al método **color()**, que cumple con la función principal de colorear los países, para ello, se recorre la lista de éstos y se llama al método privado **colourCountries()**, donde se inicializa la variable **usedColour** a false, que se empleará para cuando se haya utilizado un color, y se recorre la lista de colores, posteriormente se recorren las fronteras, si el color de la iteración del primer bucle coincide con el color de la frontera, **usedColour** pasará a true y se saldrá del bucle de fronteras. En caso de que no se haya usado el color, una vez acabe la iteración se pintará con ese color el país, y si ese color no estaba en la lista de colores usados se añadirá a dicha lista (para saber los colores empleados y el número de éstos), una vez hecho esto se sale del bucle y se vuelve a marcar **usedColour** como false.*

Una vez termine de pintar todos los países, muestra la lista de éstos y los colores empleados.

B) ¿Cuántos colores has necesitado para resolver el problema dado?

Se han necesitado 5 colores.

C) ¿Podría cambiar el número de colores necesario si utilizas un orden diferente para procesar los países del fichero de la entrada?

Si, podría cambiar, reduciéndose así el número de colores usados a 4.

D) ¿Cuántos colores utilizarías, como mucho, en una solución óptima?

Sería necesario utilizar 4 colores como mucho para conseguir la solución más óptima.

E) ¿Cuál es la complejidad temporal de tu algoritmo? Explícala brevemente.

La complejidad del Algoritmo es $O(n^2)$, debido a lo siguiente:

En el primer bucle se recorre la lista de países completa, lo que hace que la complejidad ya sea O(n), sin embargo, en el segundo bucle, al no recorrer la lista de países completa su complejidad no llega a O(n), y, por último, en el tercer bucle se vuelve a recorrer la lista de países, y, como ya se explicó anteriormente, esto presenta una complejidad temporal de O(n).

```
/**
 * Método que actualiza las referencias entre países
 */
private void updateReferences() {
    List<Country> countriesList = new ArrayList<Country>(countriesMap.keySet());
    int i = 0;
    for (Country mainCountry : countriesList) { //Recorrer la lista de países
        List<Country> frontiersList =  countriesMap.get(mainCountry);   //Cargar la lista de fronteras
        for (Country frontier : frontiersList) {    //Recorrer fronteras de cada país
            for (Country country : countriesList) { //Recorrer la lista de países 2
                if (frontier.getName().equals(country.getName())) { //Si tienen el mismo nombre ->
                    frontiersList.set(i, country);      //Se cambia la referencia
                }
            }
            i++;
        }
        i=0;
    }
}
```

Con esto, al estar anidados los bucles, hace que la complejidad del algoritmo sea $O(n^2)$, cabe mencionar que, en el resto de métodos, la complejidad alcanza como mucho O(n), por lo que, la final corresponde a la ya mencionada.