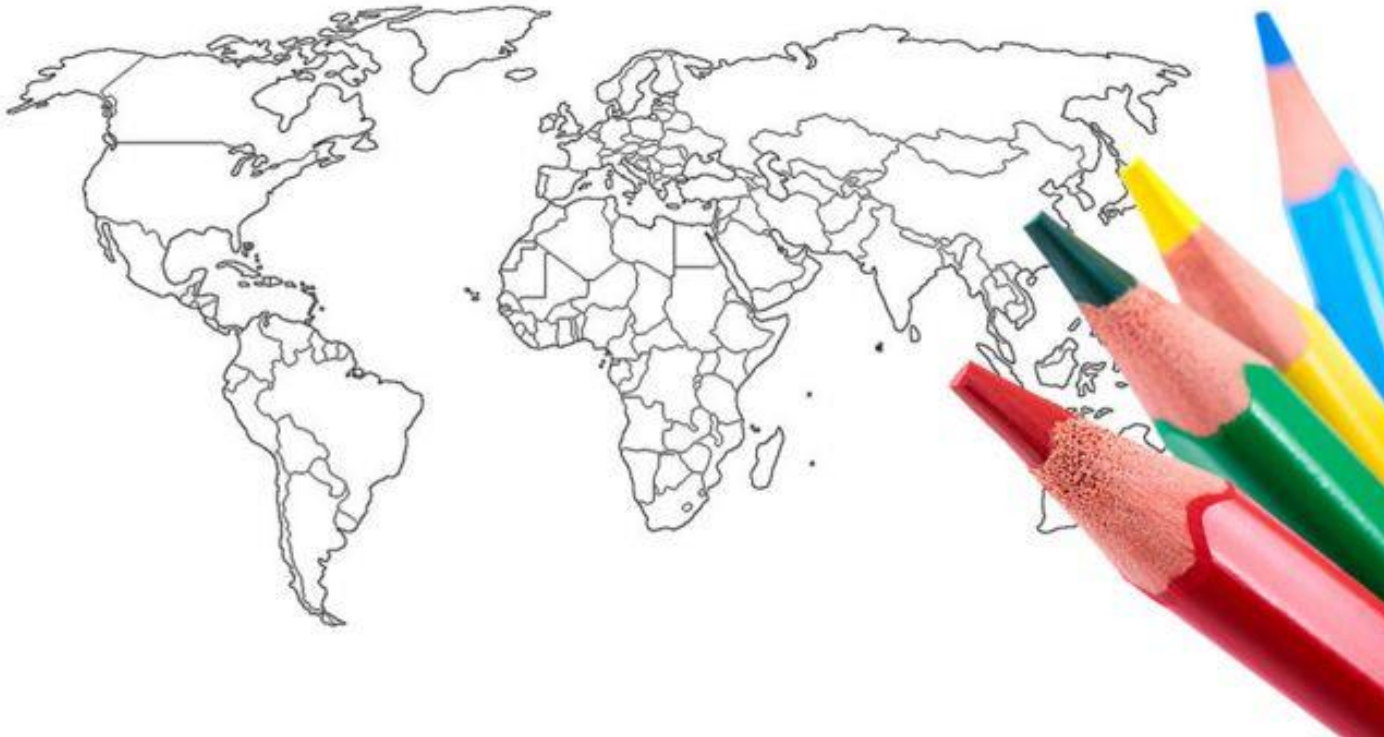


# Coloreado de mapas

# Coloreado de mapas

Dado un mapa (conjunto de elementos con conexiones entre sí), determinar el número mínimo de colores necesarios para asignarle uno a cada elemento, de un total de  $N$ , sin que dos elementos adyacentes queden del mismo color

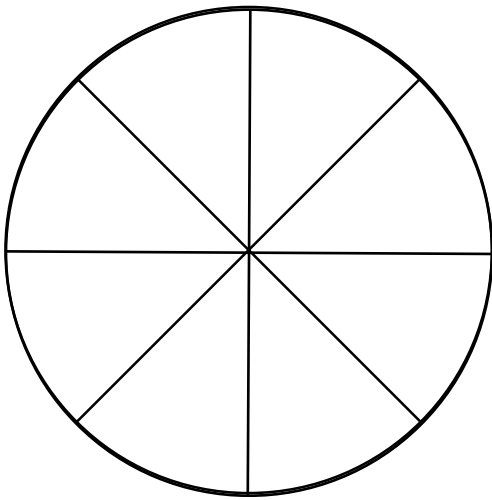


En 1879, el matemático inglés Alfred Kempe argumentó que todos los mapas provienen de un grupo finito, o "conjunto inevitable", de mapas "simplificados" y estos se podían colorear con solo 4 colores.

# Coloreado de mapas

Asumiendo la teoría de Kempe como cierta, ¿cuántas formas diferentes habría de pintar un mapa de  $N$  elementos?  $4^N$

Ahora, considerando adyacencias con perímetro cero como el mostrado en la figura, ¿cuántas formas habría?  $N^N$



# Solución mediante búsqueda exhaustiva

¿Cuál es la eficiencia de este algoritmo?  $N^{N+2}$

```
colors = 1
while colors ≤ N:
    for i = 0 to N:
        ci = 0
    do:
        if true:
            if process(c):
                print colors
                exit
        while nextCombination(c)
    colors += 1
```

```
process(c):
    for i = 0 to N-2
        for j = i+1 to N-1
            if ci = cj and adjacent(i,j):
                return false
    return true
```