```python
# C:\Users\moham\Desktop\Code de projet Ferromagnétisme.py
import time
import numpy as np
import matplotlib.pyplot as plt
import scipy
import numpy.random as random


###


def InitReseau(n): #On crée un réseau de spin up
    reseau = np.ones([n,n])
    return reseau

def energietotale(Spin,n): # On calcule l'énergie du réseau
    energie = 0
    for k in range(n):
        for j in range(n):
            energie += Spin[k][j]
    return -2*Je*energie

def Aimantationtotale(Spin,n): # On calcule l'aimantation du réeseau
    aimantation = 0
    for k in range(n):
        for j in range(n):
            aimantation += Spin[k][j]

    return aimantation /n**2


def DeltaE(Spin,k,j,n):
    E = Spin[k-1][j] + Spin[(k+1)%n][j] + Spin[k][j-1] + Spin[k]
[(j+1)%n]

    return 2*Je*E*Spin[k][j]

def DeltaM(Spin,k,j,n):
    return -2*Spin[k][j]/n**2


def EvolutionSpin(Spin,n,Temp): # On fait évoluer un spin
aléatoirement dans le réseau
    k=random.randint(0,n-1)
    j=random.randint(0,n-1)

    dE = DeltaE(Spin,k,j,n)
    dM = DeltaM(Spin,k,j,n)


    if random.random() < np.exp(-dE/(Kb*Temp))/(1+np.exp(-
dE/(Kb*Temp))):
        Spin[k][j] = -Spin[k][j]
    else:
        dE = dM = 0
    return [Spin,dE,dM]


###

```

```
057| N = 50
058| Reseau = InitReseau(N)
059| Gen = 10000
060|
061|
062| Kb = 1
063| Je = 1
064| T = 5
065|
066| En = [energietotale(Reseau,N)]
067| Ai= [Aimantationtotale(Reseau,N)]
068|
069|
070| for k in range(Gen):
071|     Reseau,dE,dM = EvolutionSpin(Reseau,N,T)
072|     En.append(En[k]+dE)
073|     Ai.append(Ai[k]+dM)
074|
075|
076|
077| ## Affichage
078|
079| plt.figure('Energie moyenne')
080| plt.grid(True)
081| plt.xlabel ('Generation')
082| plt.ylabel('Energie')
083| plt.plot(En)
084|
085|
086| plt.figure('Aimantation moyenne')
087| plt.grid(True)
088| plt.xlabel('Generation')
089| plt.ylabel('Aimantation moyenne')
090| plt.plot(Ai)
091| plt.show()
092|
093|
094|
095|
096| ###
097| Temperature = np.linspace(T,0.1,1001)
098| ##
099| Energie = []
100| Aimantation = []
101| for t in Temperature :
102|
103|     for k in range(Gen):
104|         Spin,dE,dM = EvolutionSpin(Reseau,N,t)
105|         if k == 0:
106|             mi = Aimantationtotale(Spin,N)
107|             ei= energietotale(Spin,N)
108|
109|         ei = ei + dE
110|         mi= mi + dM
111|
112|     Energie.append(ei)
113|     Aimantation.append(mi)
114|
115| ##
116|
```

```
117| plt.figure('Aimantation')
118| plt.grid(True)
119| plt.xlabel('Temperature')
120| plt.ylabel('Aimantation')
121| plt.plot(Temperature,Aimantation)
122|
123| plt.figure('Energie')
124| plt.grid(True)
125| plt.xlabel('Temperature')
126| plt.ylabel('Energie')
127| plt.plot(Temperature,Energie)
128|
129|
130| plt.show()
131|
132| ## Tentative de mapping pour mettre en avant les domaines de Weiss,
mais c'est un échec, à mettre en place pour la soutenance !
133| def couleur(x,y):
134|     couleur=[]
135|     for k in range(len(x)):
136|         for j in range (len(y)):
137|
138|             if x[k][j] == 1:
139|                 couleur.append("r")
140|             else:
141|                 couleur.append("b")
142|     return couleur
143| ##
144| Spin = EvolutionSpin(Reseau,len(Reseau),T)[0]
145| for k in range(Gen):
146|     Spin = EvolutionSpin(Spin,N,T)[0]
147|
148| ##
149|
150|
151| plt.imshow(Spin)
152| plt.show()
153|
154|
155|
156|
157|
158|
```