```java
// TSP
public class RecuitTSP {
        /*******************************************************/
        /*  Parametres du recuit                               */
        private static final int nbTransitions=2000;
        private static final double alpha=0.995;
        private static final boolean minimisation=true;
        /*******************************************************/
        /* dimension du probleme                               */
        private static final int DIMENSION=100;

        // principe d'acceptation
        private boolean accept(double yi, double yj, double T, boolean min) {
            boolean isAccepted = false;
            double dE = yj-yi;
            double proba = Math.exp(-Math.abs(dE)/T);
            double tirage = Math.random();

            if(min) {
                if(dE<0) isAccepted = true;
                else if (tirage <= proba) isAccepted=true;//acceptation frequente si T grand car alors proba proche de 1
            } else {// opt en maximisation
                if(dE>0) isAccepted = true;
                else if (tirage <= proba) isAccepted=true;
            }
            return isAccepted;
        }
    //**************************************
    //        Heat Up
    //*************************************
    public double heatUpLoop()
    {   //HeatUp heat = new HeatUp();
        int acceptCount = 0;
        double yi=0,yj;
        double T=0.01, tauxAccept=0.0;
        EtatTSP xi=new EtatTSP(DIMENSION);
        do {
            acceptCount=0;
            for (int i=0; i< nbTransitions;i++)
            {
                //generation d'un point de l'espace d'etat
                xi.initAleatEtat();
                yi=xi.calculCritere();

                //generation d'un voisin
                xi.genererVoisin();
                yj=xi.calculCritere();

                if (accept(yi,yj,T,minimisation)) acceptCount++;
                tauxAccept=(double)acceptCount/(double)nbTransitions;
            }
            T=T*1.1;
            System.out.println("T= " + T + " tauxAccept= " + tauxAccept + " currentCost= " + yi);
        } while(tauxAccept<0.8);
    return T;
    }
    //*****************************************
    //              COOLING
    //*****************************************
    public EtatTSP coolingLoop(double Tinit)
    {   //HeatUp heat = new HeatUp();
        double yi=0.0,yj=0.0;
        double T=Tinit;
        EtatTSP xi=new EtatTSP(DIMENSION);

        xi.initAleatEtat();
        yi=xi.calculCritere();
        do {
            for (int i=0; i< nbTransitions;i++)
            {
                xi.genererVoisin();
                yj=xi.calculCritere();
                if (accept(yi,yj,T,minimisation)){
                    yi=yj;
                }
                else
                {
                    xi.comeBack();
                }
            }
            T=T*alpha;
            System.out.println("T= " + T + " valeur critere " + yi);
            System.out.println(xi.afficherEtat());
        } while(T>0.0001*Tinit);
        return xi;
    }
    //*******************************************
    //           MAIN
    //*******************************************
    public static void main( String args[] )
    {
        double temperature;
        RecuitTSP monRecuit = new RecuitTSP();
        //generation des donnees
        System.out.println("***********************Generation des donnee
***************");
        Data.genererVillesCercle(DIMENSION);
        //Data.genererVilles(DIMENSION);

        System.out.println("***************************Chauffage ***************");
        temperature = monRecuit.heatUpLoop();
        System.out.println("=======================Refroidissement =============");
        monRecuit.coolingLoop(temperature);
    }//end main
}//End class HeatUp2
```