



**Benemérita Universidad  
Autónoma de Puebla**

**Facultad de Ciencias  
De la Computación**

# Inteligencia Artificial

**Luis Rene Marcial Castillo**  
Docente

**Andrés Cruz Chipol**  
Alumno

**201817637**  
Matricula



```

function [ruta,costo] = agtap(Npob,pmut,gen,nameFile)
clc
ruta = [];costo = [];
%cargar base de datos
datos = load(nameFile);

s = size(datos); %tamaño de los datos
p = s(1,1); %almacenar el numero de ciudades
inicial = []; %vector inicial de las ciudades
for i = 1:p
    inicial(i) = i;
end
inicial; %verificar vector inicial
%almacenar la poblacion Inicial con permutaciones del vector inicial
pobInicial = [];

for i=1:Npob
    i
    pobInicial(i,:) = permutacion(inicial);
end

% CICLO DE HAMILTON
% CAMBIAAR
pobAuxiliar = pobInicial;
tam = size(pobInicial);
tamVector = tam(1,1);
A = zeros(tamVector,1);
%%Creacion de la poblacion Inicial con la evaluacion
for i = 1: tamVector
    suma = 0;
    for j= 1:(p - 1)
        ki = pobInicial(i,j); ke = pobInicial(i,j+1);
        d1 = (datos(ke,2) - datos(ki,2))^2; d2 = (datos(ke,3) -
datos(ki,3))^2;
        res = sqrt(d1 + d2 );
        suma = suma + res;
    end
    ki = pobInicial(i,p); ke = pobInicial(i,1);
    d1 = (datos(ke,2) - datos(ki,2))^2; d2 = (datos(ke,3) -
datos(ki,3))^2;
    res = sqrt(d1 + d2 );
    suma = suma + res;
    A(i,1) = suma;
end
%%Uniendo los resultados a la matriz
format('long','g');
pobAuxiliar;
pobFinal = [pobAuxiliar A];
%%Ordenar la Matriz
pobFinal = sortrows(pobFinal,p+1);
%%Cruza Y Mutacion
for i= 1:gen
    i
    %Cruze y mutacion
    ij = 1;
    tamPob = size(pobFinal);

```

```

np = tamPob(1,1);
for jj = 1 : np/2
    %%Generacion de Pc1 y Pc2
    pc1 = randi([1 p-1],1);
    pc2 = randi([1 p-1],1);
    if(pc1 > pc2)
        aux = pc1;
        pc1 = pc2;
        pc2 = aux;
    end
    while pc1 == pc2
        pc1 = randi([1 p-1],1);
        pc2 = randi([1 p-1],1);
        if(pc1 > pc2)
            aux = pc1;
            pc1 = pc2;
            pc2 = aux;
        end
    end

    %%Invertir las parejas
    h1 = pobFinal(1,1:p);
    h2 = pobFinal(ij,1:p);

    intervaloUno = h1(pc1:pc2);
    intervaloDos = h2(pc1:pc2);
    %%invertir parte media
    h1(pc1:pc2) = intervaloDos;
    h2(pc1:pc2) = intervaloUno;

    %%cambiar los elementos
    t = size(intervaloDos);
    t(1,2);
    %    Encontrar indice
    for ii=1: t(1,2)
        inh1 = find(h1 == intervaloDos(ii),1);
        inh2 = find(h2 == intervaloUno(ii),1);

        aux = h1(inh1);
        h1(inh1) = h2(inh2);
        h2(inh2) = aux;
    end

    %%Mutacion
    if rand(1) <= pmut
        pd2 = randi([1 p],1);
        pd1 = randi([1 p],1);

        pos1 = h1(pd1);
        pos2 = h1(pd2);
        h1(pd1) = pos2;
        h1(pd2) = pos1;

        pd2 = randi([1 p],1);

```

```

        pd1 = randi([1 p],1);

        pos1 = h2(pd1);
        pos2 = h2(pd2);
        h2(pd1) = pos2;
        h2(pd2) = pos1;
    end

    %%Evaluacion de los hijos
    Hn = [];
    Hn = [Hn;h1;h2];
    ij = ij + 2;
end

tam = size(Hn);
tamVector = tam(1,1);
D = zeros(tamVector,1);
for i = 1: tamVector
    suma = 0;
    for j= 1:(p - 1)
        ki = Hn(i,j);  ke = Hn(i,j+1);
        d1 = (datos(ke,2) - datos(ki,2))^2; d2 = (datos(ke,3) -
datos(ki,3))^2;
        res = sqrt(d1 + d2 );
        suma = suma + res;
    end
    ki = Hn(i,p);  ke = Hn(i,1);
    d1 = (datos(ke,2) - datos(ki,2))^2; d2 = (datos(ke,3) -
datos(ki,3))^2;
    res = sqrt(d1 + d2 );
    suma = suma + res;
    D(i,1) = suma;
end
hnd = [Hn D];
pobFinal = [pobFinal;hnd];
pobFinal = sortrows(pobFinal,p+1);
pobFinal = pobFinal(1:8,:);
ruta = pobFinal(1,1:p);
costo = pobFinal(1,p+1:p+1);
end

costo

%GRAFICO DE MATLAB
vectorRutaX = [];
vectorRutaY = [];
for i=1:p
    s = ruta(1,i);
    vectorRutaX = [vectorRutaX datos(s,2)]; %posicion X
    vectorRutaY = [vectorRutaY datos(s,3)]; %posicion Y
end

```

```
x = vectorRutaX;  
y = vectorRutaY;  
plot(x,y, '-s', 'MarkerSize', 10, ...  
      'MarkerEdgeColor', 'red', ...  
      'MarkerFaceColor', [1 .6 .6]) ;
```

# Sahara

i =

12000

costo =

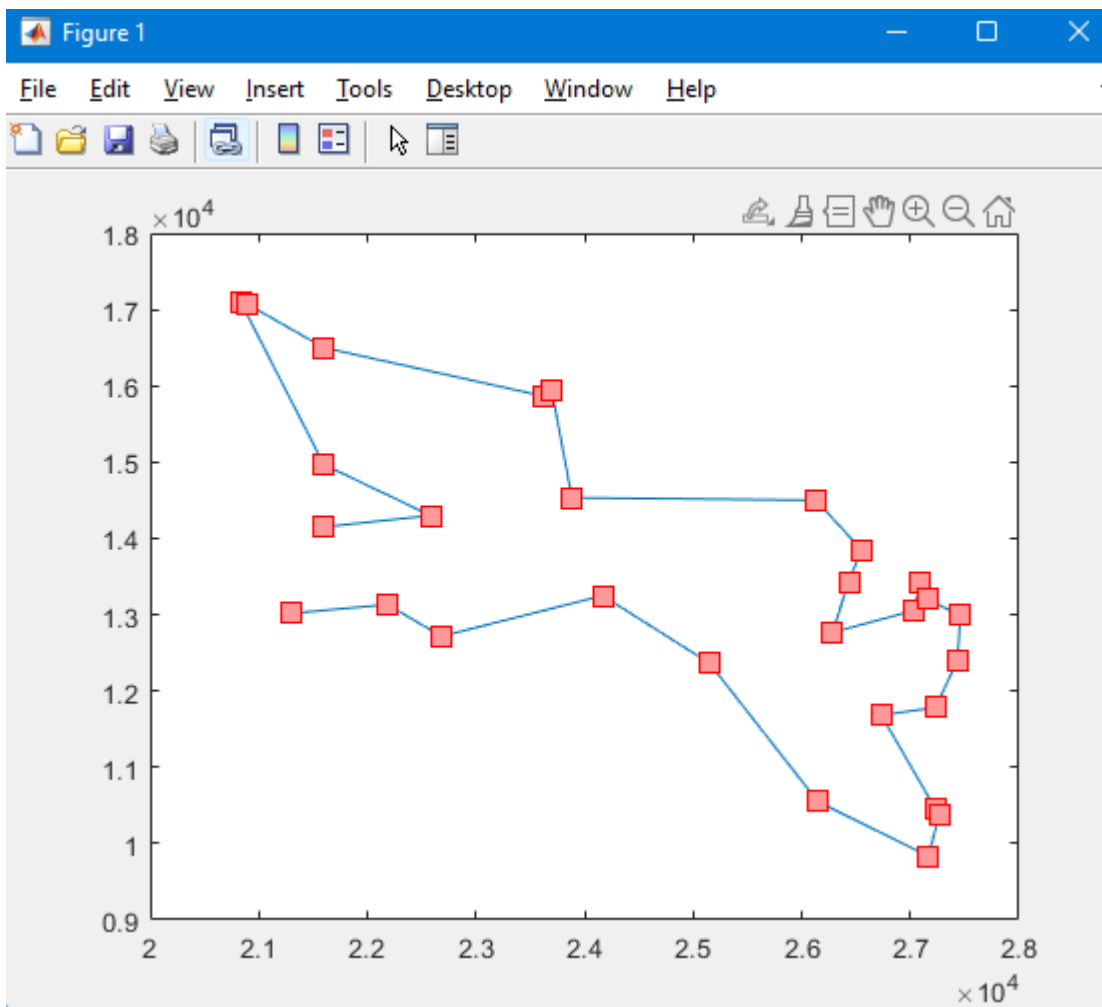
27601.1737744938

ans =

4 8 5 1 2 6 10 11 12 15 19 18 17 21 22 23 29 28 26 20 25 27 24 16 14 13 9 7 3

Ruta :

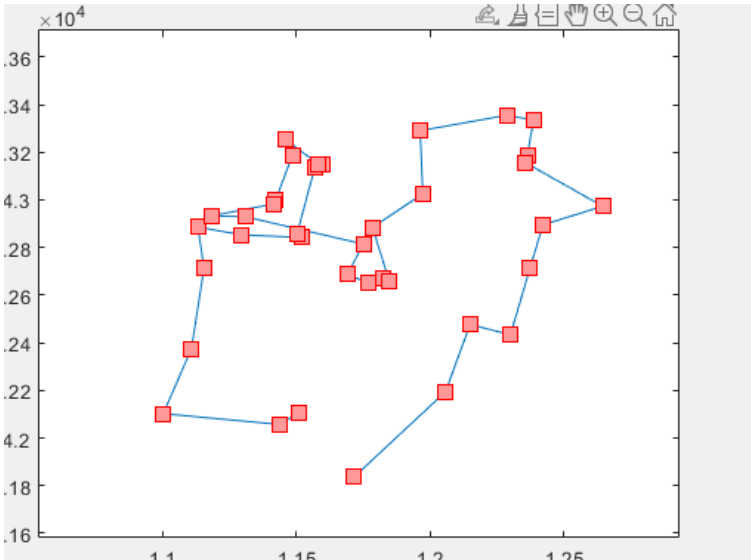
4 8 5 1 2 6 10 11 12 15 19 18 17 21 22 23 29 28 26 20 25 27  
24 16 14 13 9 7 3



# Djibouti

```
i =  
  
14000  
  
costo =  
  
7357.03134138185|  
  
ans =  
  
Columns 1 through 33  
14 10 1 2 4 3 6 15 13 16 18 19 17 11 12 9 8 5 7 22 20 23 25 26 24 28 27 31 36  
  
Columns 34 through 38  
35 32 30 29 21
```

Ruta: 14 10 1 2 4 3 6 15 13 16 18 19 17 11 12 9 8 5 7 22 20 23 25 26 24 28 27 31 36  
34 33 38 37 35 32 30 29 21



# Yemen

i =

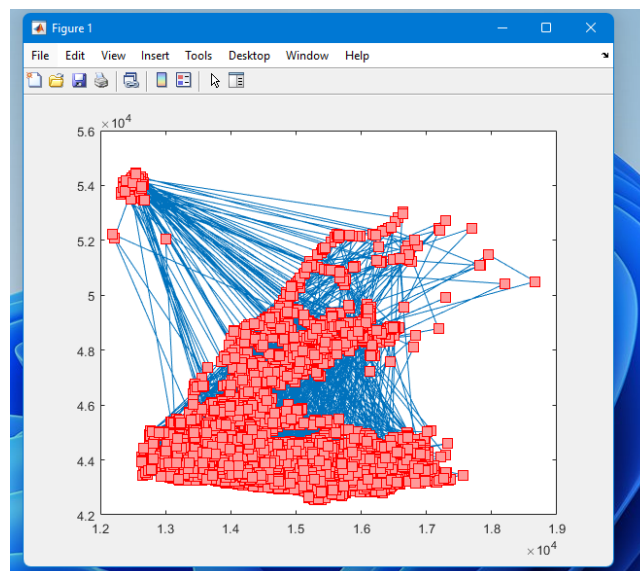
150000

costo =

6951400.47363287

1372	2679	2121	4003	2577	4427	3136	3673	3134	3263	3051	3389	4263
Columns 7,541 through 7,553												
3895	5015	4880	6771	7593	7250	7096	871	1128	7312	7327	7175	7611
Columns 7,554 through 7,566												
6568	3988	3748	3114	5105	5179	2915	5582	3038	3424	3478	4627	4652
Columns 7,567 through 7,579												
5302	2256	2810	2660	6191	5491	2150	1538	1376	1249	433	592	697
Columns 7,580 through 7,592												
1061	1285	3810	3545	3793	3824	2922	2650	5507	4969	5569	5910	6794
Columns 7,593 through 7,605												
7204	7348	5697	2815	2798	934	724	154	2306	4959	7051	7246	7016
Columns 7,606 through 7,618												
6267	6310	5325	3795	1130	443	144	152	470	1726	1518	3330	4011
Columns 7,619 through 7,631												
3405	1964	1763	1111	505	325	167	139	3544	4674	7265	7236	7371
Columns 7,632 through 7,644												
4811	4882	5124	2284	2800	2803	3476	3783	1300	2495	2552	2122	1790
Columns 7,645 through 7,657												
1510	2381	7140	7552	7375	6714	3309	2799	2836	1378	1410	2263	3982
Columns 7,658 through 7,663												
3177	3520	4781	6029	5493	5047							

Npob = 200, pmut=0.75 gen 150000





# Nicaragua:

i =

300000

costo =

1363891.90034499

I

Columns 7,541 through 7,553

3895	5015	4880	6771	7593	7250	7096	871	1128	7312	7327	7175	7611
------	------	------	------	------	------	------	-----	------	------	------	------	------

Columns 7,554 through 7,566

6568	3988	3748	3114	5105	5179	2915	5582	3038	3424	3478	4827	4652
------	------	------	------	------	------	------	------	------	------	------	------	------

Columns 7,567 through 7,579

5302	2256	2810	2660	6191	5491	2150	1538	1376	1249	433	592	697
------	------	------	------	------	------	------	------	------	------	-----	-----	-----

Columns 7,580 through 7,592

1061	1285	3810	3545	3793	3824	2922	2650	5507	4969	5569	5910	6794
------	------	------	------	------	------	------	------	------	------	------	------	------

Columns 7,593 through 7,605

7204	7348	5697	2815	2798	934	724	154	2306	4959	7051	7246	7016
------	------	------	------	------	-----	-----	-----	------	------	------	------	------

Columns 7,606 through 7,618

6267	6310	5325	3795	1130	443	144	152	470	1726	1518	3330	4011
------	------	------	------	------	-----	-----	-----	-----	------	------	------	------

Columns 7,619 through 7,631

3405	1964	1763	1111	505	325	167	139	3544	4674	7265	7236	7371
------	------	------	------	-----	-----	-----	-----	------	------	------	------	------

Columns 7,632 through 7,644

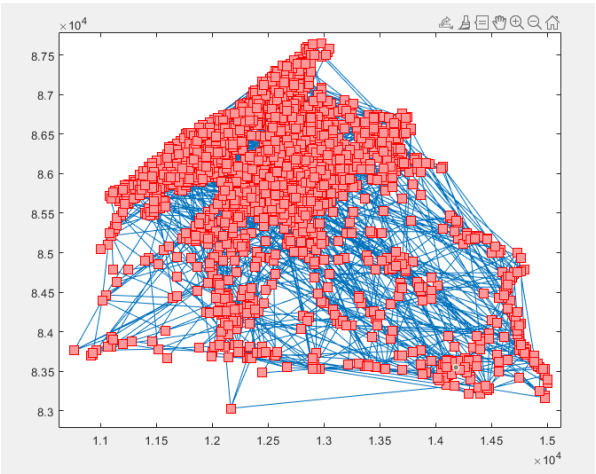
4811	4882	5124	2284	2800	2803	3476	3783	1300	2495	2552	2122	1790
------	------	------	------	------	------	------	------	------	------	------	------	------

Columns 7,645 through 7,657

1510	2381	7140	7552	7375	6714	3309	2799	2836	1378	1410	2263	3982
------	------	------	------	------	------	------	------	------	------	------	------	------

Columns 7,658 through 7,663

3177	3520	4781	6029	5493	5047
------	------	------	------	------	------



Npob = 200 pmut = 0.75 gen = 300000