



Programación Evolutiva

Facultad de Informática

Curso 2015/2016

Práctica 2.

El objetivo de esta práctica es implementar un algoritmo evolutivo para resolver el clásico problema del viajante de comercio. Es un problema de optimización combinatoria que nos va a permitir probar la mayoría de los conceptos vistos en clase.

Problema: Dadas n ciudades etiquetadas de 1 a N y las distancias entre unas y otras d_{ij} ($i, j \in 1..N$), se trata de calcular el recorrido más corto que pasa por todas las ciudades y que comienza y termine en la misma ciudad.

En nuestro caso concreto el viajante sale de Madrid y tiene que recorrer 27 ciudades por España, pasando por todas ellas, sin repetir ciudades, buscando el camino más corto y acabando el recorrido en Madrid.

La aplicación debe mostrar las gráficas de evolución y el resultado (recorrido) final obtenido. La interfaz de usuario debe permitir elegir cualquiera de los parámetros (Tamaño de la población, Número de generaciones, Porcentaje de cruces, Porcentaje de mutaciones, Elitismo) y además, permitirá seleccionar lo siguiente:

- ☐ El método de selección: Ruleta, Torneo, Ranking, Restos, Truncamiento.
- ☐ El método de cruce: PMX, OX, Variantes de OX, Ciclos (CX), Recombinación de rutas (ERX), Codificación Ordinal, y algún método propio.
- ☐ El método de mutación: Inserción, Intercambio, Inversión, Heurística, y algún método propio
- ☐ Cualquiera de las mejoras vistas en clase (escalado, contractividad,...).

Opcional: Incluir un estudio de los resultados obtenidos al variar ciertos parámetros dentro de un rango de valores. Para ello la interfaz debe permitir fijar un parámetro entre un valor inferior y uno superior y lanzar diferentes ejecuciones para diferentes valores de los parámetros.

Documentación a entregar

- ☐ Hay que enviar al campus virtual antes del **25 de abril a las 12:00** un archivo comprimido con el código java de la aplicación (**proyecto en Eclipse o NetBeans**) cuyo nombre se corresponda con el nombre del grupo y las siglas **P2**, por ejemplo **G01P2**.
- ☐ En el archivo comprimido se incluirá una breve memoria que contenga el estudio de las gráficas y resultado obtenidos con cada función. Aquí se valorarán las conclusiones y observaciones que se consideren interesantes respecto al resultado obtenido, a las mejoras utilizadas y a los métodos propios
- ☐ El día de corrección será en la sesión de Laboratorio del **25 de abril** y deberán estar presentes los dos integrantes del grupo. La práctica deberán conocerla a fondo los dos componentes del grupo pues se harán preguntas a ambos indistintamente.

	Distancias kilométricas																																																			
	Albacete																																																			
Alicante	171	Alicante																																																		
Almería	369	294	Almería																																																	
Avila	366	537	663	Avila																																																
Badajoz	525	696	604	318	Badajoz																																															
Barcelona	540	515	809	717	1022	Barcelona																																														
Bilbao	646	817	958	401	694	620	Bilbao																																													
Burgos	488	659	800	243	536	583	158	Burgos																																												
Cáceres	504	675	651	229	89	918	605	447	Cáceres																																											
Cádiz	617	688	484	618	342	1284	1058	900	369	Cádiz																																										
Castellón	256	231	525	532	805	284	607	524	701	873	Castellón																																									
Ciudad Real	207	378	407	256	318	811	585	427	324	464	463	Ciudad Real																																								
Córdoba	354	525	332	457	272	908	795	637	319	263	610	201	Córdoba																																							
A Coruña	860	1031	1172	538	772	1118	644	535	683	1072	1026	799	995	A Coruña																																						
Cuenca	142	313	511	282	555	562	562	404	451	708	305	244	445	776	Cuenca																																					
Gerona	640	615	909	817	1122	100	720	683	1018	1384	384	911	1008	1218	662	Gerona																																				
Granada	363	353	166	534	438	868	829	671	485	335	584	278	166	1043	479	968	Granada																																			
Guadalajara	309	480	621	173	459	563	396	238	355	721	396	248	458	667	486	663	492	Guadalajara																																		
Huelva	506	703	516	552	251	1140	939	781	323	219	856	433	232	1006	677	1240	350	690	Huelva																																	
Huesca	495	570	830	490	798	274	322	359	694	1060	355	587	797	905	406	374	831	339	1029	Huesca																																
Jaén	264	415	228	435	376	804	730	572	423	367	520	179	104	944	380	904	99	393	336	732	Jaén																															
León	584	855	896	255	496	784	359	201	407	796	725	511	733	334	500	884	761	391	730	560	668	León																														
Lérida	515	490	802	558	866	156	464	427	762	1128	259	655	865	973	472	256	861	407	1097	118	779	628	Lérida																													
Logroño	578	653	899	358	676	468	152	115	595	999	455	526	736	650	464	568	770	278	968	244	671	316	312	Logroño																												
Lugo	762	933	1074	440	674	1020	546	437	585	974	928	696	897	98	678	1120	945	569	908	807	846	236	875	352	Lugo																											
Madrid	251	422	563	115	401	621	395	237	297	663	417	190	400	609	167	721	434	58	632	397	335	333	465	336	511	Madrid																										
Málaga	473	482	219	644	436	997	939	781	506	265	713	388	187	1153	615	1097	129	602	313	941	209	877	1009	880	1055	544	Málaga																									
Murcia	150	75	219	516	675	590	796	638	654	613	306	357	444	1010	292	690	278	459	628	611	340	734	583	694	912	401	407	Murcia																								

```
static int[][] DIST = {
    {},
    {251},
    {422, 171},
    {563, 369, 294},
    {115, 366, 537, 663},
    {401, 525, 696, 604, 318},
    {621, 540, 515, 809, 717, 1022},
    {395, 646, 817, 958, 401, 694, 620},
    {237, 488, 659, 800, 243, 536, 583, 158},
    {297, 504, 675, 651, 229, 89, 918, 605, 447},
    {663, 617, 688, 484, 618, 342, 1284, 1058, 900, 369},
    {417, 256, 231, 525, 532, 805, 284, 607, 524, 701, 873},
    {190, 207, 378, 407, 256, 318, 811, 585, 427, 324, 464, 463},
    {400, 354, 525, 332, 457, 272, 908, 795, 637, 319, 263, 610, 201},
    {609, 860, 1031, 1172, 538, 772, 1118, 644, 535, 683, 1072, 1026, 799, 995},
    {167, 142, 313, 511, 282, 555, 562, 562, 404, 451, 708, 305, 244, 445, 776},
    {721, 640, 615, 909, 817, 1122, 100, 720, 683, 1018, 1384, 384, 911, 1008, 1218, 662},
    {434, 363, 353, 166, 534, 438, 868, 829, 671, 485, 335, 584, 278, 166, 1043, 479, 968},
    {58, 309, 480, 621, 173, 459, 563, 396, 238, 355, 721, 396, 248, 458, 667, 486, 663, 492},
    {632, 506, 703, 516, 552, 251, 1140, 939, 781, 323, 219, 856, 433, 232, 1006, 677, 1240, 350, 690},
    {397, 495, 570, 830, 490, 798, 274, 322, 359, 694, 1060, 355, 587, 797, 905, 406, 374, 831, 339, 1029},
    {335, 264, 415, 228, 435, 376, 804, 730, 572, 423, 367, 520, 179, 104, 944, 380, 904, 99, 393, 336, 732},
    {333, 584, 855, 896, 255, 496, 784, 359, 201, 407, 796, 725, 511, 733, 334, 500, 884, 761, 391, 730, 560, 668},
    {465, 515, 490, 802, 558, 866, 156, 464, 427, 762, 1128, 259, 655, 865, 973, 472, 256, 861, 407, 1097, 118, 779, 628},
    {336, 578, 653, 899, 358, 676, 468, 152, 115, 595, 999, 455, 526, 736, 650, 464, 568, 770, 278, 968, 244, 671, 316, 312},
    {511, 762, 933, 1074, 440, 674, 1020, 546, 437, 585, 974, 928, 696, 897, 98, 678, 1120, 945, 569, 908, 807, 846, 236, 875, 352},
    {544, 473, 482, 219, 644, 436, 997, 939, 781, 506, 265, 713, 388, 187, 1153, 615, 1097, 129, 602, 313, 941, 209, 877, 1009, 880, 1055},
    {401, 150, 75, 219, 516, 675, 590, 796, 638, 654, 613, 306, 357, 444, 1010, 292, 690, 278, 459, 628, 611, 340, 734, 583, 694, 912, 401, 407}
};
```