CECS 552 Programming Assignment 4

Applying Simulated Annealing to Political Rediscricting

Dr. Ebert

November 8th

City Redistricting

A representative from an anonymous city in Norway has been charged with the task of reforming the city's voting districts. Currently there are 66 districts, and each one sends a representative to serve on the city council. However, the council voted to amend its constitution so that, as of 2019, the council will only consist of 28 members, one from each newly formed district. The current districts have been deemed unfair because the populations vary significantly from district to district. Thus, the plan is to create new districts, where each new district is the result of merging one or more of the old disctricts. Moreover, the goal is to minimize the quantity

$$S = \sum_{i=1}^{28} (P_i - \mu)^2,$$

where P_i is the population of the i th new district, and $\mu = P/28$, where P is the total city population. However, there is one constraint that each district must satisfy. Namely, the old districts that comprise each new district must form a connected graph, where the graph vertices are the old districts, and two vertices/districts are adjacent iff their boundaries intersect at at least one point.

The following tables provides the population of each district, and, for each district, a list of bordering districts.

District	Population	District	Population	District	Population
1	2,992	23	2,987	45	3,023
2	2,032	24	2,031	46	2,973
3	3,021	25	2,969	47	2,024
4	1,973	26	3,028	48	5,976
5	2,020	27	2,991	49	1,978
6	2,977	28	3,001	50	1,975
7	2,003	29	1,993	51	1,969
8	3,004	30	2,010	52	2,978
9	2,985	31	2,995	53	2,030
10	3,024	32	2,979	54	2,002
11	3,032	33	2,008	55	2,971
12	3,004	34	2,010	56	1,991
13	3,020	35	3,027	57	2,997
14	1,980	36	2,991	58	2,024
15	2,026	37	1,974	59	2,990
16	2,008	38	2,018	60	1,993
17	1,980	39	1,979	61	2,009
18	1,984	40	3,028	62	3,004
19	1,978	41	2,020	63	3,026
20	2,008	42	2,983	64	2,984
21	2,008	43	3,004	65	3,011
22	1,991	44	1,970	66	2,993

Dist.	Neighbors	Dist.	Neighbors	Dist.	Neighbors
1	32,23,45,12,28,38	23	42,45,1,32	45	42,14,56,46,10,12,1,23
2	47,36,5,61,55	24	13,6,55,61,14,7,48	46	56,31,62,27,57,10,45
3	52,53	25	8,19,47,55,58,66	47	55,25,58,40,36,2
4	16,34,38,9,54	26	38,22,49,9	48	13,24,7,64,18
5	55,2,36,56,14,61	27	46,31,62,10,57	49	9,26,22
6	15,55,24,13,37	28	1,12,43,38,59	50	22,38
7	48,24,61,14,42,64	29	60,35,65,34,44,39,63	51	13,17,18,30,33,37,41
8	19,25,66,58	30	37,41,51	52	3,17,21,33,53
9	4,38,26,22,49	31	56,36,62,27,46	53	3,17,52,60
10	45,46,27,12,57,43	32	65,42,23,1,38	54	4,16
11	18,64,42,65,63	33	17,21,41,51,52	55	15,19,25,58,47,2,5,61,24,13
12	1,45,10,28,59,43,57	34	44,29,65,38,4,16	56	14,61,5,2,36,31,46,45,42
13	37,15,6,24,48,18,51,55	35	60,18,63,65,29	57	12,10,46,27,43
14	7,24,61,5,56,42,45	36	2,47,58,40,5,56,31	58	8,25,36,40,47,55,66
15	6,19,37,55,13	37	6,13,15,19,30,51	59	28,12,43,38
16	20,44,34,4,54	38	22,34,65,32,1,28,59,50,26,9,4	60	17,18,20,29,35,39,53,63
17	18,33,51,52,53,60	39	20,60,29,44	61	24,55,2,5,56,14,7
18	17,51,13,48,64,11,63,35,60	40	36,47,58,66	62	27,31,46
19	8,15,25,37	41	21,30,33,51	63	60,18,11,65,35,29
20	60,39,44,16	42	64,7,14,45,23,32,65,11,56	64	18,48,7,42,65,11
21	33,41,52	43	12,10,57,28,59	65	29,35,63,11,64,42,32,38,34
22	9,26,38,50,49	44	20,39,29,34,16	66	8,25,58,40

Use simulated annealing to find an approximate solution to the re-districting problem. Use the state space for which each state $x = (s_1, \ldots, s_{28})$ consists of a collection of 28 pairwise disjoint sets of old districts, in which each old-district set is connected (when the districts are viewed as vertices of a graph), and each old district belongs to exactly one of the 28 sets. Moreover, assume that y is a neighbor of x iff y can be obtained from x by moving a district d from one set s to another set s', without disconnecting s, and such that d is adjancent to some district in s'.

Program Options

1. The user inputs a list of districts, and the program prints the connected components of the subgraph formed by these discricts and their adjacencies. For example, if the user inputs 3,11,18,21,52, then the program should return $\{3, 21, 52\}$, $\{11, 18\}$.

2. The user inputs the name of a text file, and the program reads in a state x from that file, and prints all legal neighbors of x, and provides the S-values for each neighbor and x. Note: the file is formatted so that each line represents a single set of old districts whose ids are separated by commas.

3. The user inputs a time limit (in seconds), and the program runs simulated annealing for that amount of time, and returns the best state found during the search, along with the population of each new district, and its associated S-value.