Try the algorithm above with two objective functions $y_1 = (x-2)^2$ and $y_2 = (x-4)^2$ as follows:

1. *Create 20 10-bit binary chromosomes, assuming each chromosome represent x-cordinate ranges from 0 to 6 with $(0000\cdots00)$ and $(1111\cdots11)$ being corresponding to 0 and 6, respectively.*
2. *Calculate y_1 and y_2 for each of 20 x's represented by these 20 chromosomes.*
2. *Select individuals uniformly from population.*
3. *Perform crossover and mutation to create a child.*
4. *Calculate the rank of the new child.*
5. *Find the individual in the entire population that is most similar to the child. Replace that individual with the new child if the child's ranking is better, or if the child dominates it.*

   1

6. *Update the ranking of the population if the child has been inserted.*
7. *Perform steps 2-6 according to the population size.*
8. *If the stop criterion is not met go to step 2 and start a new generation.*

Chromosome	x	y1	y2	Rank
0111110111	2,95014663	0,90277861	1,1021921	8
1110000010	5,26686217	10,6723884	1,60493976	0
1101111110	5,24340176	10,519655	1,54604794	1
1000001100	3,07331378	1,15200248	0,85874734	8
1011101111	4,40469208	5,78254401	0,16377568	3
1101111101	5,23753666	10,4816436	1,53149698	2
0000110101	0,31085044	2,85322624	13,6098245	3
0000000101	0,02932551	3,88355793	15,7662559	0
0111101011	2,8797654	0,77398715	1,25492557	8
0110100110	2,47507331	0,22569465	2,3254014	5
0000011011	0,15835777	3,3916461	14,758215	1
1010001101	3,82991202	3,34857801	0,02892992	6
0111000101	2,6568915	0,43150644	1,80394045	5
0000101101	0,26392962	3,01394037	13,9582219	2
0111000101	2,6568915	0,43150644	1,80394045	5
0111101000	2,86217009	0,74333726	1,29465691	8
1000100100	3,21407625	1,47398113	0,61767615	8
1011000000	4,12903226	4,53277836	0,01664932	4
0001011110	0,55131965	2,09867476	11,8933962	4
0000101011	0,25219941	3,05480689	14,0460092	5