Jakub Pleban

Dokumentacja projektu z Analizy Algorytmów.

Temat: Poszukiwanie największego okręgu wpisanego w wielokąt.

Problem:

Nie w każdy wielokąt możliwe jest wpisanie idealnie okręgu, ale jest możliwe znalezienie takiego środka dla okręgu i takiego promienia, że co najwyżej może istnieć okrąg o takim samym promieniu (przykład prostokąt, który nie jest kwadratem). Warunkiem na to, aby znaleźć taki okrąg jest jego styczność z co najmniej trzema elementami tego wielokąta (kombinacja wierzchołków oraz boków wielokąta).

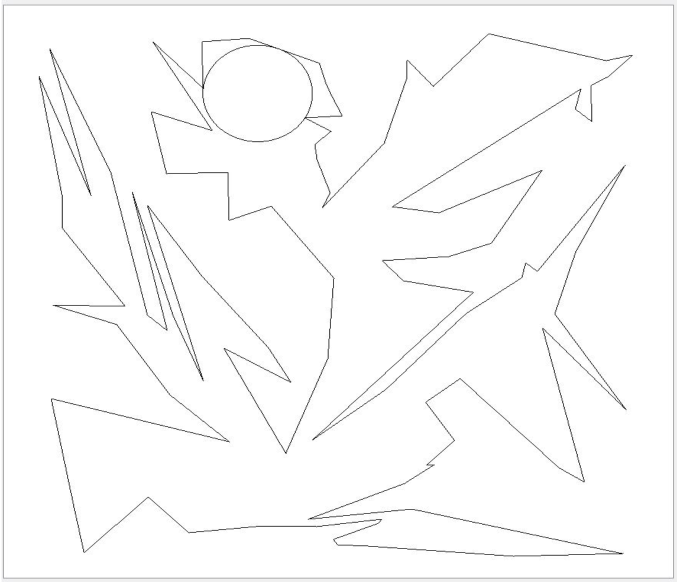
Wniosek:

Należy znaleźć wewnątrz wielokąta punkty, które będą równo odległe od przynajmniej trzech elementów wielokąta i wyznaczyć jeden, dla którego promień będzie największy oraz okrąg utworzony nie będzie przecinać boków wielokąta.

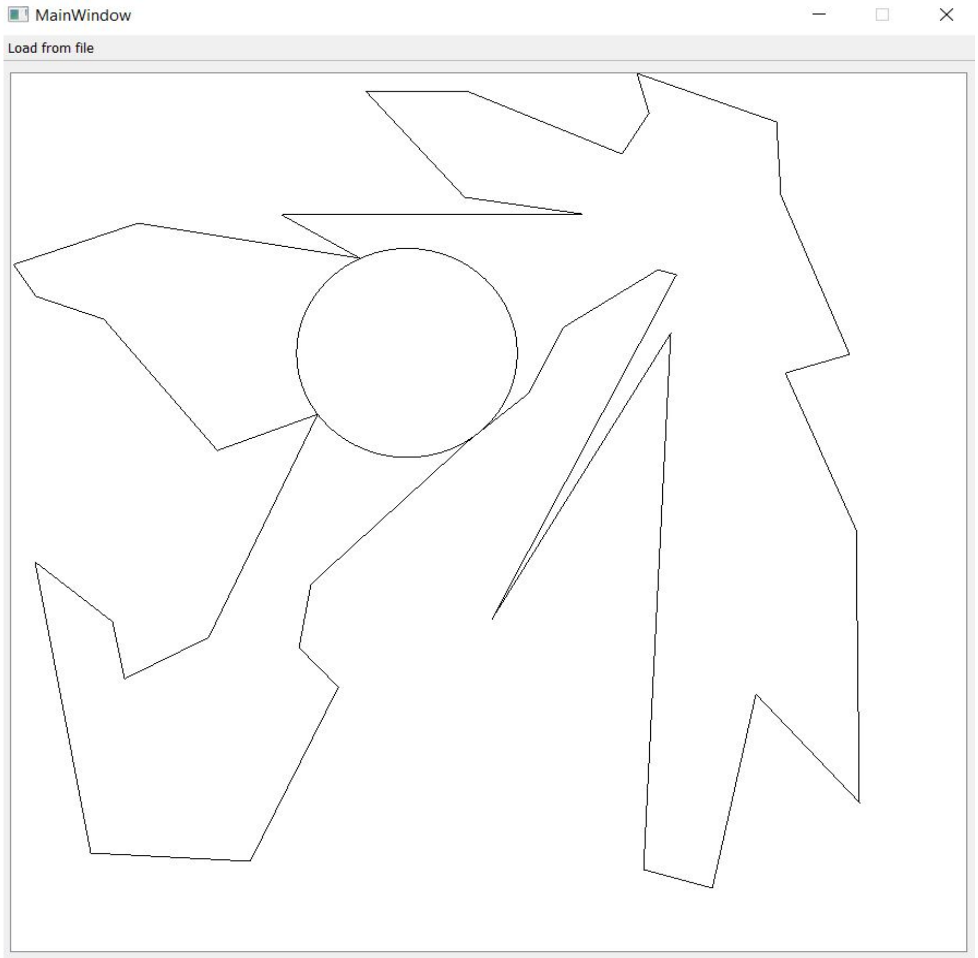
Rozwiązanie:

W rozwiązaniu brutalnym będziemy wyznaczać punkty, które będą równo odległe od trzech różnych składowych wielokąta. Do tego będzie trzeba wyznaczyć kombinacje tych składowych i znalezienie potencjalnego środka okręgu oraz jego promienia (złożoność O(n^3)), a następnie przeszukać te środki w celu znalezienia takiego, który będzie leżeć w środku wielokąta, nie przecinającego boków wielokąta (złożoność O(n)), o największym promieniu. W sumie złożoność rozwiązania wyniesie O(n^4).

Przykład rozwiązania wyświetlony za pomocą aplikacji w Qt (N = 100, czas generacji około 3min):



Inny przykład:



Przykłady działania algorytmy z mierzeniem czasu (w ms) dla:

- N = 10, k = 10, s = 1, r = 5

n	t(n)	q(n)
10	31	1.05327
11	45	1.04429
12	67	1.09781
13	84	0.999274
14	114	1.00826
15	149	1
16	190	0.985037
17	228	0.927509
18	294	0.95156
19	361	0.941178

- N = 10, k = 10, s = 2, r = 10

n	t(n)	q(n)
10	30	1.0984
12	61	1.07707
14	111	1.05791
16	189	1.05589
18	305	1.06377
20	437	1
22	636	0.994042
24	850	0.938021
26	1131	0.906166
28	1516	0.903037

– - N = 10, k = 20, s = 1, r = 10

n	t(n)	q(n)
10	30	1.03896
11	45	1.06444
12	64	1.06889
13	89	1.07918
14	114	1.02771
15	154	1.0535
16	190	1.00404
17	231	0.957843
18	299	0.986414
19	383	1.0178
20	462	1
21	519	0.924205
22	639	0.944687
23	727	0.899707
24	849	0.886218
25	1015	0.899879
26	1206	0.91397
27	1342	0.874532
28	1535	0.864877
29	1721	0.842688

- N = 10, k = 20, s = 2, r = 10

n	t(n)	q(n)
10	32	1.26009
12	67	1.27233
14	120	1.23004
16	187	1.1236
18	322	1.20786
20	442	1.08781
22	675	1.13465
24	869	1.0314
26	1212	1.04438
28	1606	1.02888
30	2057	1
32	2615	0.982025
34	3269	0.963274
36	4111	0.963803
38	5232	0.988062
40	6132	0.94322
42	7576	0.958724
44	9050	0.950799
46	10469	0.920714
48	12417	0.921091

Wniosek: Algorytm ze złożonością n^4 nie jest szczególnie optymalny, ale nadal należy do klasy P, dobrze radzi sobie z wyznaczaniem środka okręgu w wielokątach do około ~40-50 wierzchołków.