

# GEOMETRIA OBLICZENIOWA

## Zestaw 4 – Triangulacja Delaunay

Napisać program wyznaczający triangulację Delaunay. Program ma mieć następujące możliwości:

1. losowe generowanie zadanej liczby punktów wejściowych leżących w  $[0, 1]^2$ ,
2. wczytywanie punktów z pliku tekstowego (w pierwszej linii pliku podana jest liczba punktów  $n$ , a w kolejnych  $n$  liniach współrzędne punktów),
3. rysowania punktów wraz z ich triangulacją.

Do wyznaczanie triangulacji Delaunay należy wykorzystać algorytm 1.

---

### Algorytm 1: Triangulacja Delaunay

---

**Dane:** Zbiór  $P$  zawierający  $n \in \mathbb{N}$  punktów przestrzeni  $\mathbb{R}^2$ .

**Wynik:** Triangulacja Delaunay  $\mathcal{T}$  zbioru  $P$ .

- 1 Niech  $(x_l, y_l), (x_u, y_u)$  będą lewym dolnym i prawym górnym rogiem prostokąta zawierającego punkty z  $P$
  - 2  $(x_b, y_b) = ((x_l + x_u)/2, (y_l + y_u)/2)$
  - 3  $d = 100 \cdot \max\{|x_u - x_l|, |y_u - y_l|\}$
  - 4  $p_{-1} = (x_b - d, y_b - d), p_{-2} = (x_b, y_b + d), p_{-3} = (x_b + d, y_b - d)$
  - 5 Inicjuj  $\mathcal{T}$  jako triangulację zawierającą pojedynczy trójkąt o wierzchołkach  $p_{-1}, p_{-2}, p_{-3}$
  - 6 Oblicz losową permutację punktów  $p_1, p_2, \dots, p_n \in P$
  - 7 **for**  $r = 1$  **to**  $n$  **do**
    - 8 Znajdź trójkąt  $T \in \mathcal{T}$  zawierający  $p_r$ ; oznaczmy jego wierzchołki przez  $p_i, p_j, p_k$
    - 9 **if**  $p_r$  leży we wnętrzu trójkąta  $T$  **then**
      - 10 Dodaj krawędzie z  $p_r$  do trzech wierzchołków  $p_i, p_j, p_k$  dzieląc w ten sposób  $T$  na trzy trójkąty
      - 11 `LEGALIZEEDGE`( $p_r, \overline{p_i p_j}, \mathcal{T}$ )
      - 12 `LEGALIZEEDGE`( $p_r, \overline{p_j p_k}, \mathcal{T}$ )
      - 13 `LEGALIZEEDGE`( $p_r, \overline{p_k p_i}, \mathcal{T}$ )
    - 14 **else**
      - 15 Dodaj krawędzie z  $p_r$  do  $p_k$  i do trzeciego wierzchołka  $p_l$  drugiego trójkąta sąsiadującego z krawędzią  $\overline{p_i p_j}$ , dzieląc w ten sposób dwa trójkąty sąsiadujące z  $\overline{p_i p_j}$  na cztery trójkąty
      - 16 `LEGALIZEEDGE`( $p_r, \overline{p_i p_l}, \mathcal{T}$ )
      - 17 `LEGALIZEEDGE`( $p_r, \overline{p_l p_j}, \mathcal{T}$ )
      - 18 `LEGALIZEEDGE`( $p_r, \overline{p_j p_k}, \mathcal{T}$ )
      - 19 `LEGALIZEEDGE`( $p_r, \overline{p_k p_i}, \mathcal{T}$ )
  - 20 Usuń z  $\mathcal{T}$  punkty  $p_{-1}, p_{-2}, p_{-3}$  wraz ze wszystkimi incydentnymi z nimi krawędziami
  - 21 **return**  $\mathcal{T}$
-

---

**Algorytm 2:** Sprawdzanie czy punkt leży wewnątrz trójkąta

---

**Dane:** Punkt  $p \in \mathbb{R}^2$ , który chcemy sprawdzić czy leży w trójkącie o wierzchołkach  $v_1, v_2, v_3 \in \mathbb{R}^2$ .

**Wynik:** FALSE – punkt leży poza trójkątem, TRUE – punkt leży wewnątrz lub na trójkącie.

```
1  $\mathbf{w}_1 = p - v_1$ 
2  $\mathbf{w}_2 = p - v_2$ 
3  $\mathbf{w}_3 = v_2 - v_1$ 
4  $\mathbf{w}_4 = v_3 - v_2$ 
5  $s_1 = \mathbf{w}_{1y}\mathbf{w}_{3x} - \mathbf{w}_{1x}\mathbf{w}_{3y}$ 
6  $s_2 = \mathbf{w}_{2y}\mathbf{w}_{4x} - \mathbf{w}_{2x}\mathbf{w}_{4y}$ 
7 if  $s_1$  i  $s_2$  mają przeciwne znaki then
8   | return FALSE
9  $\mathbf{w}_5 = p - v_3$ 
10  $\mathbf{w}_6 = v_1 - v_3$ 
11  $s_3 = \mathbf{w}_{5y}\mathbf{w}_{6x} - \mathbf{w}_{5x}\mathbf{w}_{6y}$ 
12 if  $s_1$  i  $s_3$  mają przeciwne znaki then
13   | return FALSE
14 return TRUE
```

---

---

**Algorytm 3:** LEGALIZEEDGE

---

**Dane:**  $p_r$  jest wstawianym punktem,  $\mathcal{T}$  jest triangulacją,  $\overline{p_i p_j}$  jest krawędzią w  $\mathcal{T}$ , która może wymagać przekroczenia.

```
1 LEGALIZEEDGE( $p_r, \overline{p_i p_j}, \mathcal{T}$ )
2   | if  $\overline{p_i p_j}$  jest nielegalna then
3     | Niech  $T$  będzie trójkątem o wierzchołkach  $p_i, p_j, p_k$  sąsiadującym z trójkątem
4     |    $T'$  o wierzchołkach  $p_r, p_i, p_j$  wzdłuż krawędzi  $\overline{p_i p_j}$ 
5     |   Zastąp  $\overline{p_i p_j}$  przez  $\overline{p_r p_k}$ .
6     |   LEGALIZEEDGE( $p_r, \overline{p_i p_k}, \mathcal{T}$ )
7     |   LEGALIZEEDGE( $p_r, \overline{p_k p_j}, \mathcal{T}$ )
```

---

---

**Algorytm 4:** Sprawdzanie czy krawędź jest nielegalna

---

**Dane:**  $p_i, p_j$  końce krawędzi.

**Wynik:** FALSE – krawędź jest nielegalna, TRUE – krawędź jest legalna.

```
1 if  $i, j < 0$  then
2   | return TRUE
3 Niech  $p_k, p_l$  będą pozostałymi wierzchołkami trójkątów współdzielących krawędź
4    $\overline{p_i p_j}$ 
5 if  $p_l$  leży wewnątrz okręgu opisanego na trójkącie o wierzchołkach  $p_i, p_j, p_k$  then
6   | return FALSE
7 else
8   | return TRUE
```

---

---

**Algorytm 5:** Sprawdzanie czy punkt leży wewnątrz okręgu opisanego na trójkącie

---

**Dane:** Trójkąt  $T$  o wierzchołkach  $v_1, v_2, v_3 \in \mathbb{R}^2$  oraz punkt  $p \in \mathbb{R}^2$ , który testujemy.

**Wynik:** 0 – punkt leży w okręgu, 1 – punkt leży poza okręgiem, 2 – punkt leży na okręgu.

$$1 \quad \mathcal{O} = \begin{vmatrix} v_{1x} & v_{1y} & 1 \\ v_{2x} & v_{2y} & 1 \\ v_{3x} & v_{3y} & 1 \end{vmatrix}$$
$$2 \quad \mathcal{C} = \begin{vmatrix} v_{1x} & v_{1y} & v_{1x}^2 + v_{1y}^2 & 1 \\ v_{2x} & v_{2y} & v_{2x}^2 + v_{2y}^2 & 1 \\ v_{3x} & v_{3y} & v_{3x}^2 + v_{3y}^2 & 1 \\ p_x & p_y & p_x^2 + p_y^2 & 1 \end{vmatrix}$$

```
3  if  $\mathcal{O} > 0$  then
4  |   if  $\mathcal{C} > 0$  then
5  |   |   return 0
6  |   else
7  |   |   if  $\mathcal{C} < 0$  then
8  |   |   |   return 1
9  |   |   else
10 |   |   |   return 2
11 else
12 |   if  $\mathcal{C} < 0$  then
13 |   |   return 0
14 |   else
15 |   |   if  $\mathcal{C} > 0$  then
16 |   |   |   return 1
17 |   |   else
18 |   |   |   return 2
```

---

## Literatura

- [1] de Berg, M., van Kreveld, M., Overmars, M., Schwarzkopf, O.: Geometria obliczeniowa algorytmy i zastosowania. WNT, Warszawa, (2007)
- [2] Guibas, L.J., Knuth, D.E., Sharir, M.: Randomized Incremental Construction of Delaunay and Voronoi Diagrams. *Algorithmica* 7(1), 381-413, (1992)
- [3] Hjelle, Ø., Dæhlen, M.: *Triangulations and Applications*. Springer, Heidelberg, (2006)