

Лабораторна робота № 6

Діаграма Вороного, триангуляція Делоне і побудова лінійної опуклої оболонки на площині

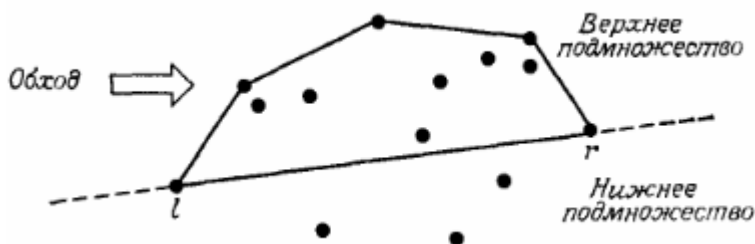
1. Реалізувати алгоритм Форчуна побудови діаграми Вороного (fortune.doc) і його поточну візуалізацію (5 балів).
2. Здійснити триангуляцію Делоне (1 бал), тобто побудувати діаграму, двоїсту до діаграми Вороного.
3. Реалізувати алгоритм Кейла-Кіркпатрика побудови лінійної опуклої оболонки  $N$  точок на площині з цілочисельними координатами ( $1 \leq x_i \leq m_x; 1 \leq y_i \leq m_y; i = \overline{1, N}$ ). Складність алгоритму лінійна:  $N + M$ , де  $M = \max\{m_x, m_y\}$ . (1 бал)

Ідея:

- 1) відсортувати задані точки за  $y$ -координатою кишеньковим сортуванням; при цьому для кожного  $i = \overline{1, m_y}$  визначити крайні ліву та праву точки серед заданих точок,  $y$ -координата яких дорівнює  $i$ . В результаті буде отримано 2 масиви  $A_l, A_r$  «лівих» та «правих» точок, які містять в собі вершини лівої та правої частини опуклої оболонки.
  - 2) зробити обхід (напр. знизу вгору) по  $A_l$ ; при цьому відсіяти точки, які не є вершинами лівої частини опуклої оболонки (умова: в результаті при обході  $A_l$  знизу вгору поворот має здійснюватися завжди НАПРАВО). Аналогічно зробити обхід  $A_r$ . Результат вивести на екран.
4. Реалізувати мішаний алгоритм Ендрю та Джарвіса побудови лінійної опуклої оболонки  $N$  точок на площині ( $1 \leq x_i \leq m_x; 1 \leq y_i \leq m_y; i = \overline{1, N}$ ). Складність алгоритму  $N^2$ . (1 бал)

Ідея:

- 1) відсортувати множину заданих точок за  $x$ -координатою; при цьому визначити крайню ліву та крайню праву точки.



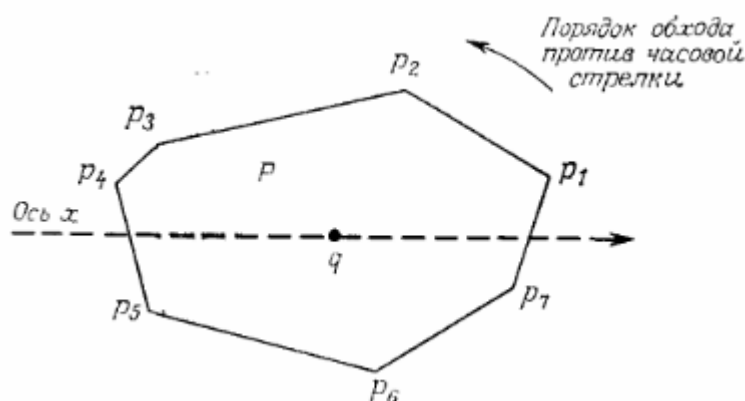
2) послідовно визначити ребра верхньої частини оболонки, використовуючи правило: відрізок  $\gamma$ , що визначається двома точками є ребром опуклої оболонки тоді і тільки тоді, коли всі інші точки заданої множини розташовані на  $\gamma$  або з одного боку від нього. Аналогічно побудувати множину ребер нижньої частини опуклої оболонки.

Результат вивести на екран.

5. Реалізувати алгоритм Грехема побудови лінійної опуклої оболонки  $N$  точок на площині ( $1 \leq x_i \leq m_x; 1 \leq y_i \leq m_y; i = \overline{1, N}$ ). Складність алгоритму:  $N \log N$ . (1 бал)

Ідея:

- 1) відсортувати точки по полярному куту з центром в точці  $q$  як показано на рисунку, де  $q$ -внутрішня точка лінійної оболонки, яка вважається відомою. При цьому можна скористатися правилом: кут точки  $p_2$  більше кута точки  $p_1 \Leftrightarrow$ , коли трикутник  $qp_1p_2$  має строго додатно орієнтовану площу;



- 2) зробити повторний обхід відсортованого масиву (списку) точок. При цьому видалити зайві точки так, як показано на рис.

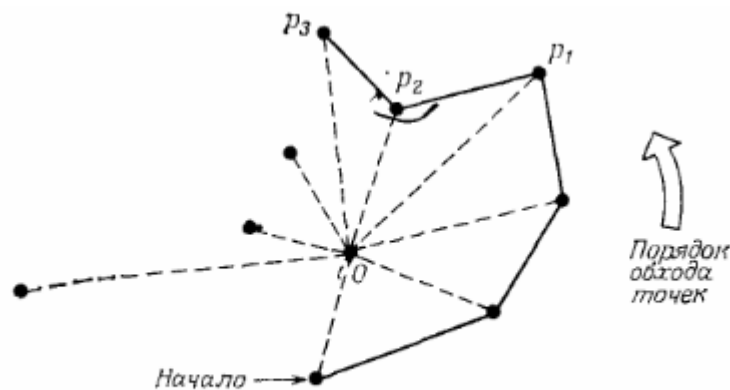


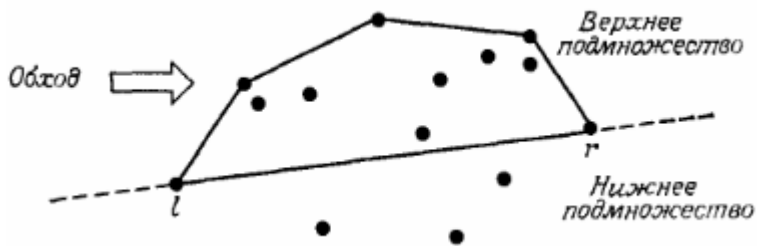
Рис. Начало обхода точек в методе Грэхема. Вершина  $p_2$  удаляется, если угол  $p_1p_2p_3$  оказывается вогнутым.

Результат вивести на екран.

6. Реалізувати швидкий рекурсивний алгоритм побудови лінійної опуклої оболонки  $N$  точок на площині ( $1 \leq x_i \leq m_x; 1 \leq y_i \leq m_y; i = \overline{1, N}$ ). Складність алгоритму:  $N \log N$ . (1 бал)

Ідея:

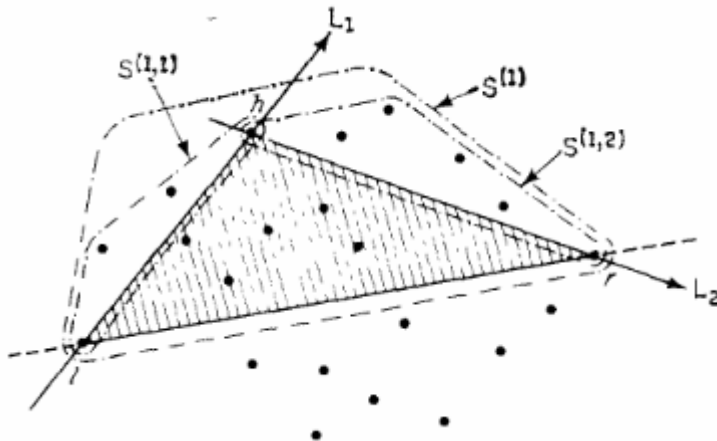
- 1) відсортувати множину заданих точок за  $x$ -координатою; при цьому визначити крайню ліву  $l$  та крайню праву  $r$  точки.



Множину  $S$ , що складена з  $N$  заданих точок розбити на дві підмножини, кожна з яких буде містити відповідно верхню та нижню ламані, з'єднання яких дає багатокутник випуклої оболонки.

Нехай при цьому  $S^{(1)}$  - підмножина точок, що розташовані вище чи на прямій  $lr$ , що сполучає крайні ліву та праву точки множини  $S$ , та  $S^{(2)}$  - підмножина точок, що розташовані нижче чи на прямій  $lr$ .

- 2) На кожному наступному кроці обробка множин, подібних до  $S^{(1)}$  чи  $S^{(2)}$  здійснюється таким чином (на прикладі множини  $S^{(1)}$  - див. рис.):



визначається точка  $h$ , для якої трикутник  $hlr$  має найбільшу площу серед всіх трикутників  $\{(plr), p \in S^{(1)}\}$ , а якщо таких точок є більше однієї, тоді вибираємо ту з них, у якої кут  $(hlr)$  більше. Точка  $h$  при цьому належить опуклій оболонці. Нехай  $L_1$ -пряма, що сполучає  $l, h$ ;  $L_2$ -пряма, що сполучає  $r, h$ . Множину  $S^{(1)}$  розбиваємо на підмножини  $S^{(1,1)}$  та  $S^{(1,2)}$  таким чином: до  $S^{(1,1)}$  відносимо точки  $S^{(1)}$ , що розташовані зліва від  $L_1$  або на ній, до  $S^{(1,2)}$  відносимо точки  $S^{(1)}$ , що розташовані справа від  $L_2$  або на ній.

Далі множини  $S^{(1,1)}, S^{(1,2)}, S^{(2,1)}, S^{(2,2)}$  передаються на наступний рівень рекурсивної обробки.