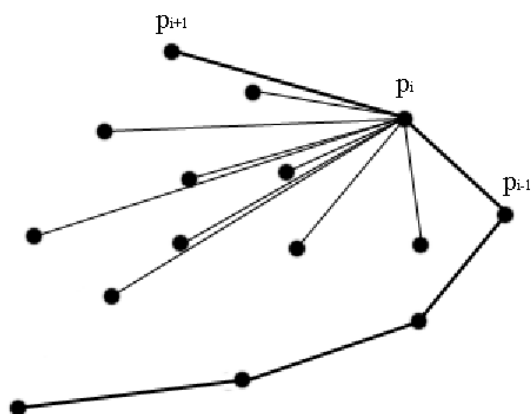
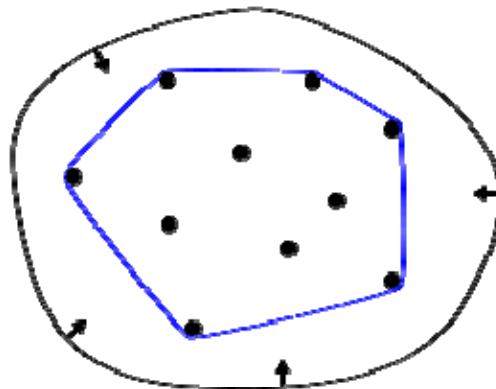


Лабораторна робота № X

«Знаходження опуклої оболонки множини точок на площині. Алгоритм Джарвіса»

Теоретичні відомості:

Опуклою оболонкою множини точок називають найменшу опуклу множину (багатокутник), що містить всі точки вихідної множини. Опуклим називають такий багатокутник, що містить у собі відрізок прямої, що з'єднує будь-які дві його вершини. Для знаходження опуклої оболонки існує чимало алгоритмів - методи Грехема, Джарвіса, Чана, метод еластичної оболонки, Quickhull та інші.



Метод Джарвіса (метод загортання подарунка) полягає у послідовному обчисленні полярних кутів між векторами, що задані попередньою знайденою точкою опуклої оболонки, поточною, та рештою множини точок, що не належать до обчисленої частини опуклої оболонки, окрім першої. Наступною точкою опуклої оболонки обирається така, що дала найменший полярний кут. За першу точку опуклої

оболонки обирають найнижчу найлівішу точку множини. Порівняння полярних кутів на площині зручно зводиться до обчислення псевдоскалярного добутку векторів, що утворені останньою обчисленою точкою опуклої оболонки та двома точками-кандидатами.

$\vec{a} \wedge \vec{b} = |a||b|\sin \theta = \begin{vmatrix} a_x & a_y \\ b_x & b_y \end{vmatrix}$. Якщо добуток від'ємний - то обирається друга точка. Якщо

добуток дорівнює нулю - точки лежать на одній прямій і обирається дальша (з більшим модулем).

Схема алгоритму:

- Знайти найнижчу, найлівішу точку множини P, додати її першою до шуканої послідовності, як p[0], i=0;
- **do:**

- $p[i+1]$ - будь-яка точка множини, що не входить до шуканої послідовності, але містить першу точку (кандидат у наступні точки опуклої границі)
- для кожної точки $P[j]$ множини, що не входить до шуканої послідовності, але містить першу точку та не містить $p[i+1]$:
 - Знайти псевдоскалярний добуток між векторами $(p[i], p[i+1])$ та $(p[i], P[j])$
 - Якщо добуток від'ємний - то $p[i+1] = P[j]$
 - Якщо добуток дорівнює нулю - порівняти відстані $(p[i], p[i+1])$ та $(p[i], P[j])$. Якщо друга більша, то $p[i+1] = P[j]$
- $i++$; **while** $p[i] \neq p[0]$; - зупинитись, коли наступна обрана точка співпаде з першою.

Отримавши опуклу оболонку множини точок, можна встановити чи знаходиться окрема точка всередині неї. Для цього використовують метод підрахунку "числа обертів" (winding number). Метод полягає у тому, що з точки будують промінь, спрямований довільно. Потім для кожного ребра опуклого багатокутника, з'ясовують, чи перетинає воно цей промінь. Якщо перетинає - то перетину присвоюють коефіцієнт +1 або -1 в залежності від того, чи перетин відбувається за годинниковою стрілкою, чи проти. Напрямок перетину з'ясовують за знаком скалярного добутку між спрямовуючими векторами променя та ребра у точці перетину. Ознакою того, що точка знаходиться всередині багатокутника, буде ненульова сума отриманих коефіцієнтів.

Завдання:

1. Розробити окрему програму, що запитує у користувача максимальні та мінімальні значення по двох координатах, число точок, та генерує множину точок із випадковими значеннями координат у вказаних межах. Точки мають бути представлені у вигляді структур, що складаються із двох реальних змінних x та y . Отримана множина має бути записана до файлу у бінарному вигляді.
2. Розробити програму, що виконує такі дії:
 - a. Отримує у користувача ім'я файлу, що містить множину точок, створену попередньою програмою;
 - b. Вчитує з файлу масив структур точок.
 - c. Знаходить та друкує межі множини за координатами та координати центроїду множини (середнє значення відповідних координат точок);
 - d. Виконує обхід множини за методом Джарвіса та знаходить масив точок, що належать до опуклої оболонки множини. Друкує їх.
3. Написати функцію, яка отримує масив опуклої оболонки та довільну точку, та з'ясовує чи знаходиться точка всередині цієї границі.
4. Протестувати програму та навести у протоколі отримані результати.