

Алгоритм:

1. Із заданої точки z проводимо промінь, паралельний до осі OX у додатному напрямку (вправо).
2. Фіксуємо точки, у яких промінь перетинає ребра багатокутника P .
3. Якщо точок-перетинів парна кількість, то точка z не належить багатокутнику P . Якщо ж перетинів непарна кількість, то точка належить багатокутнику.

Особливості реалізації:

Многокутник задається масивом об'єктів - ребер. В класі "точка" зберігаються координати точки, а в класі "ребро" - дві вершини, які його задають.

Інтерфейс програми: На вхід подається 2 файли: файл, в якому описані вершини багатокутника в послідовному порядку та файл з тестовими точками. Вихід друкується в консольному вікні: положення кожної тестової точки з вхідного файлу.

1. Враховуються тільки перетини, які змінюють стани "в/за". Тобто не враховуються перетини на ребрах, які лежать на промені.

```
def intersect(self, p):
    x_1 = self.start.x
    x_2 = self.end.x
    y_1 = self.start.y
    y_2 = self.end.y

    intersect_x = (x_2 - x_1) * (p.y - y_1) / (y_2 - y_1) + x_1
    cross = Point(intersect_x, p.y)
    if self.start == cross or self.end == cross:
        return
    else:
        return intersect_x
```

2. Не враховуються проходження променя через вершини багатокутника.

3. Якщо точка z знаходиться на лівій/нижній частині границі багатокутника (на ребрі), то вона вважається такою, що належить багатокутнику. Якщо ж точка знаходиться на правій/верхній частині границі, то вона не належить багатокутнику.

4. Правила перетину ребра променем:

- Ребро, яке перетинається променем знизу вгору (upward edge) включає свою стартову точку і не включає кінцеву.
- Ребро, яке перетинається променем згори вниз (downward edge) включає свою кінцеву точку і не включає стартову.
- Горизонтальні ребра виключаються.
- Усі точки перетину на промені мають бути строго справа від точки z .

```
crossing_n = 0 # crossing number
for edge in edges:
    if edge.is_upward(point) or edge.is_downward(
        point): # if current edge is crossed by the ray starting at point
        if point.x < edge.intersect(point): # and if intersection is on the right
            crossing_n += 1
```

5. Многокутник представлений масивом ребер.

6. Перетин ребра з променем визначається таким чином: за вершинами ребра записується рівняння прямої, на якій воно лежить. Визначається перетин променя з даною прямою. Якщо точка перетину лежить в межах ребра, то перетин зараховується.

Тестові приклади для багатокутника справа:

1. (0; 0) - всередині багатокутника.
2. (2; 1) - всередині багатокутника.
3. (-1; 3) - зовні багатокутника.
4. (-2; -1) - зовні багатокутника.

Джерело: слайди лекцій.

