- **Propoziție.** *Fie $P = (p_1, p_2), Q = (q_1, q_2)$ două puncte distincte din planul $\mathbf{R}^2$, fie $R = (r_1, r_2)$ un punct arbitrar și*

$$\Delta(P, Q, R) = \begin{vmatrix} 1 & 1 & 1 \\ p_1 & q_1 & r_1 \\ p_2 & q_2 & r_2 \end{vmatrix}.$$

Atunci R este situat:
(i) pe dreapta $PQ \Leftrightarrow \Delta(P, Q, R) = 0$ ("ecuația dreptei");
(ii) "în dreapta" segmentului orientat $\overrightarrow{PQ} \Leftrightarrow \Delta(P, Q, R) < 0$;
(iii) "în stânga" segmentului orientat $\overrightarrow{PQ} \Leftrightarrow \Delta(P, Q, R) > 0$.
- **Obs.** Testul de orientare se bazează pe calculul unui polinom de gradul II ($\Delta(P, Q, R)$).

Raport- exemple

(ii) În $\mathbb{R}^3$ considerăm punctele $A = (1, 2, 3)$, $B = (2, 1, -1)$, $C = (0, 3, 7)$. Atunci punctele A, B, C sunt coliniare și avem $r(A, C, B) = -\frac{1}{2}$, $r(B, C, A) = -2$, $r(C, A, B) = 1$, $r(C, B, A) = -2$.

(iii) Fie A, B două puncte din $\mathbb{R}^n$ și $M = \frac{1}{2}A + \frac{1}{2}B$. Atunci $r(A, M, B) = 1$, $r(M, A, B) = -\frac{1}{2}$.

Graham's scan, varianta Andrew (algoritm)

Input: O mulțime de puncte $\mathcal{P}$ din $\mathbf{R}^2$.
Output: O listă $\mathcal{L}$ care conține vârfurile ce determină frontiera acoperirii convexe, parcursă în sens trigonometric.

1. Sortare lexicografică, renumerotare $P_1, P_2, \ldots, P_n$ conform ordonării
2. $\mathcal{L} \leftarrow (P_1, P_2)$
3. **for** $i \leftarrow 3$ **to** n
4.   **do** adaugă P_i la sfârșitul lui $\mathcal{L}$
5.   **while** $\mathcal{L}$ are mai mult de două puncte
           and ultimele trei nu determină un viraj la stânga
6.   **do** șterge penultimul punct
7. **return** $\mathcal{L}_i$
8. Parcurge pași analogi pentru a determina $\mathcal{L}_s$
9. Concatenează $\mathcal{L}_i$ și $\mathcal{L}_s$

Jarvis' march (algoritm)

Input: O mulțime de puncte necoliniare $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$ din $\mathbf{R}^2$ $(n \geq 3)$.
Output: O listă $\mathcal{L}$ care conține vârfurile ce determină frontiera acoperirii convexe, parcursă în sens trigonometric.

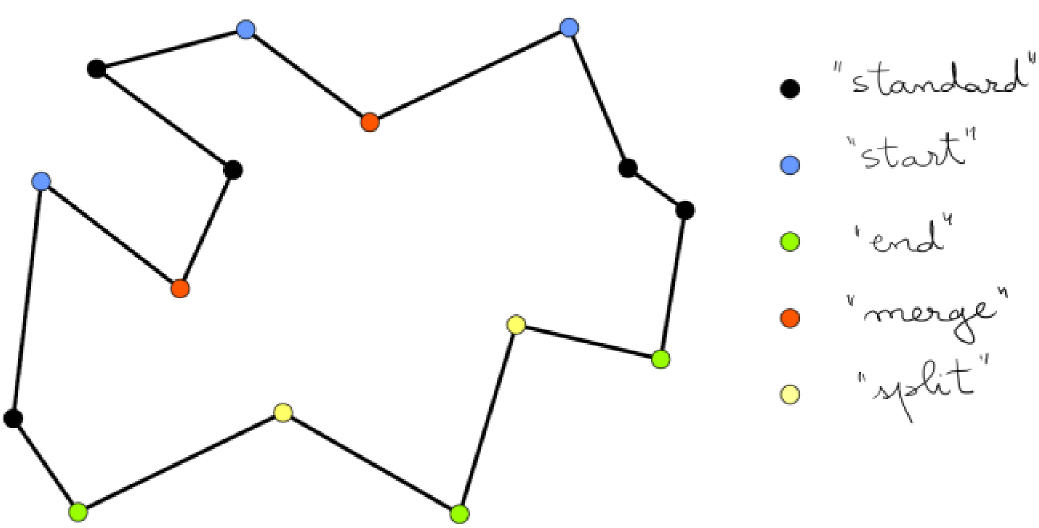
1. Determinarea unui punct din $\mathcal{P}$ care aparține frontierei (de exemplu cel mai mic, folosind ordinea lexicografică); acest punct este notat cu A_1.
2. $k \leftarrow 1$; $\mathcal{L} \leftarrow (A_1)$; *valid* $\leftarrow$ true
3. **while** *valid*$=$ true
4.   **do** alege un pivot arbitrar $S \in \mathcal{P}$, diferit de A_k
5.   **for** $i \leftarrow 1$ **to** n
6.   **do if** P_i este la dreapta muchiei orientate $A_k S$
7.   **then** $S \leftarrow P_i$
8.   **if** $S \neq A_1$
9.   **then** $k \leftarrow k + 1$;
             $A_k = S$
             adaugă A_k la $\mathcal{L}$
10.   **else** *valid* $\leftarrow$ false
11. **return** $\mathcal{L}$

Triangularea poligoanelor monotone

Input: Un poligon y-monoton $\mathcal{P}$.
Output: O triangulare a lui $\mathcal{P}$.

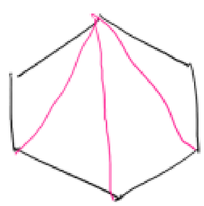
1. Lanțul vârfurilor din partea stângă și al celor din partea dreaptă sunt unite într-un singur șir, ordonat descrescător, dupa y (dacă ordonata este egală, se folosește abscisa). Fie $v_1, v_2, \ldots, v_n$ șirul ordonat.
2. Inițializează o stivă vidă $\mathcal{S}$ și inserează v_1, v_2.
3. **for** $j = 3$ **to** $n - 1$
4.   **do** **if** v_j și vârful din top al lui $\mathcal{S}$ sunt în lanțuri diferite
5.   **then** extrage toate vârfurile din $\mathcal{S}$
6.   inserează diagonale de la v_j la vf. extrase, exceptând ultimul
7.   inserează v_{j-1} și v_j în $\mathcal{S}$
8.   **else** extrage un vârf din $\mathcal{S}$
9.   extrage celelalte vârfuri din $\mathcal{S}$ dacă diagonalele formate cu v_j sunt în interiorul lui $\mathcal{P}$; inserează aceste diagonale; inserează înapoi ultimul vârf extras
10.   inserează v_j în $\mathcal{S}$
11. adaugă diagonale de la v_n la vf. stivei (exceptând primul și ultimul)

Tipuri de vârfuri



Elemente ale unei triangulări

- Dată o mulțime de puncte $\mathcal{P}$ și o triangulare $\mathcal{T}_P$ a sa: **vârfuri, muchii, triunghiuri**.
- Legătură cantitativă între aceste elemente?
- **Propoziție.** *Fie $\mathcal{P}$ o mulțime de n puncte din plan nesituate toate pe o aceeași dreaptă. Notăm cu k numărul de puncte de pe frontiera acoperirii convexe $\mathrm{Conv}(\mathcal{P})$. Orice triangulare a lui $\mathcal{P}$ are $(2n-k-2)$ triunghiuri și $(3n-k-3)$ muchii.*
- **Exemplu:** Cazul unui (unor puncte care formează un) poligon convex: un poligon convex cu n vârfuri poate fi triangulat cu $(n-2)$ triunghiuri, având $(2n-3)$ muchii.



$\underline{n = 6}$

4 △

9 muchii

Muchii ilegale

- **Criteriu numeric / analitic** pentru a testa dacă o muchie este ilegală.
  - Pentru puncte $A = (x_A, y_A), B = (x_B, y_B), C = (x_C, y_C), D = (x_D, y_D)$:

$$\Theta(A, B, C, D) = \begin{vmatrix} x_A & y_A & x_A^2 + y_A^2 & 1 \\ x_B & y_B & x_B^2 + y_B^2 & 1 \\ x_C & y_C & x_C^2 + y_C^2 & 1 \\ x_D & y_D & x_D^2 + y_D^2 & 1 \end{vmatrix}$$

  - (i) Punctele A, B, C, D sunt conciclice $\Leftrightarrow \Theta(A, B, C, D) = 0$.
    (ii) Fie A, B, C astfel ca ABC să fie un viraj la stânga. Un punct D este situat în interiorul cercului circumscris $\Delta ABC \Leftrightarrow \Theta(A, B, C, D) > 0$.

Structura unei diagrame Voronoi

- Fie $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$ o mulțime de situri (puncte) din planul $\mathbb{R}^2$.
- Celula asociată unui punct este o intersecție de semiplane:

$$\mathcal{V}(P_i) = \bigcap_{j \neq i} h(P_i, P_j),$$

  unde $h(P_i, P_j)$ este semiplanul determinat de mediatoarea segmentului $[P_i P_j]$ care conține punctul P_i. În particular: fiecare celulă este o mulțime convexă. Aplicabilitate: algoritm (lent) de determinare a diagramei Voronoi.
- Dacă toate punctele sunt coliniare, atunci diagrama Voronoi asociată $\mathrm{Vor}(\mathcal{P})$ conține $n-1$ *drepte paralele* între ele (în particular, pentru $n \geq 3$, ea nu este conexă).
- În caz contrar, diagrama este conexă, iar muchiile sale sunt fie *segmente*, fie *semidrepte* (cui corespund acestea?).
- **Propoziție.** Fie o mulțime cu n situri. Atunci, pentru diagrama Voronoi asociată au loc inegalitățile

$$n_v \leq 2n - 5, \qquad n_m \leq 3n - 6,$$

  unde n_v este numărul de vârfuri ale diagramei și n_m este numărul de muchii al acesteia.