CS112.M11.KHTN

COMPUTATIONAL GEOMETRY

GROUP 6

GROUP 6

HUNG TIEN-NGUYEN

LINH VINH-LE PHUOC

RANKING SYSTEM

Quiz

Đầu tiên, là bài quiz. Mọi người sẽ trả lời 15 câu hỏi. Điểm của nhóm bạn sẽ là tổng câu trả lời đúng của cả 2 thành viên

ACTIVE

Mỗi lần tụi mình đặt vấn đề, sau khi hết thời gian suy nghĩ thảo luận thì các nhóm sẽ raise hand để đưa ra ý kiến.

- 3 nhóm đầu tiên sẽ được **5 điểm**
- Các nhóm sau đó mỗi nhóm **3 điểm.**

PRIZE POOL

CODE: 201 083

Nickname: **Gx_*YourName***

TABLE OF CONTENTS

01

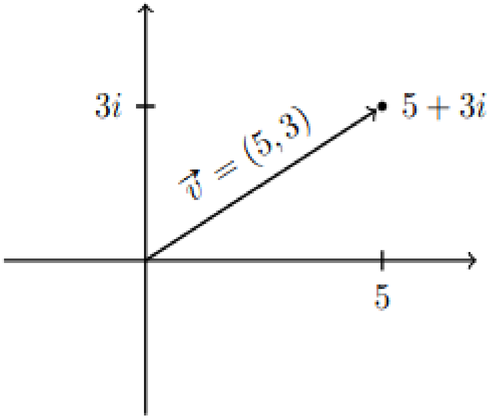
ELEMENTARY OPERATIONS

A.Điểm và Vector

1. Biểu diễn theo dạng vector:

- Với mỗi điểm A(a,b) trên mặt phẳng Oxy, ta có thể biểu diễn bằng một vector có dạng $\overrightarrow{OA} = (a, b)$

```python
class Point():
    def __init__(self, x, y):
        self.x = x
        self.y = y
```



Hình trên biểu diễn vector v = (5,3)

A.Điểm và Vector

1. Tính toán với vectơ:

- ❖ Giả sử có 2 vector u = (1, 0) và v = (3, 2).
- ❖ Các phép toán cơ bản có thể thực hiện trên 2 vector này :
  - ➢ Cộng 2 vector.
  - ➢ Trừ 2 vector.
  - ➢ Nhân vector cho một vô hướng.

```python
class Point():
  def __init__(self, x, y):
    self.x = x
    self.y = y
  def __add__(self, other):
    return Point(self.x + other.x, self.y + other.y)
  def __sub__(self, other):
    return Point(self.x - other.x, self.y - other.y)
  def __mul__(self, x):
    return Point(self.x * x, self.y * x)
```

B. Tích và góc

1. Tích vô hướng (Dot product):

❖ **Định nghĩa (Đại số):**

- Tích vô hướng của hai vector a = (a_1, a_2) và b = (b_1, b_2) được định nghĩa là:

$$a.b = a_1b_1 + a_2b_2$$

- Nếu hai vector được xem như là một **ma trận hàng** thì ta có thể viết lại:

$$a.b = ab^T$$

B. Tích và góc

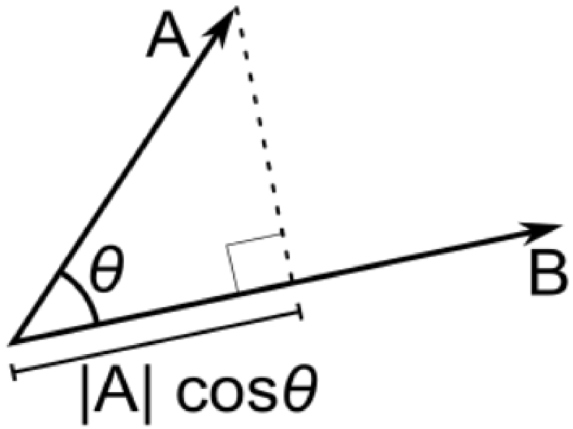
1. Tích vô hướng (Dot product):

❖ **Định nghĩa (Hình học):**

Tích giữa độ dài của vectơ thứ nhất và độ dài hình chiếu của vectơ thứ hai lên vectơ thứ nhất.

$$a.b = ||a||\;||b||cos(\theta)$$

Trong đó: $\theta \in [0, \pi]$ là góc giữa 2 vetor a và b.



B. Tích và góc

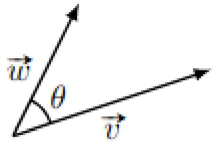
1. Tích vô hướng (Dot product):

❖ Một số tính chất thông qua tích vô hướng:

- ➢ Độ dài của vector a: $||a|| = \sqrt{a.\,a}$

- ➢ Phép chiếu của a lên b: $p = \frac{a.b}{||b||}$

- ➢ Góc giữa hai vector: $\theta = arccos(\frac{a.b}{||a||\,||b||})$



$\theta < \pi/2$

$\vec{v} \cdot \vec{w} = 5$

$\theta = \pi/2$

$\vec{v} \cdot \vec{w} = 0$

$\theta > \pi/2$

$\vec{v} \cdot \vec{w} = -5$

B. Tích và góc

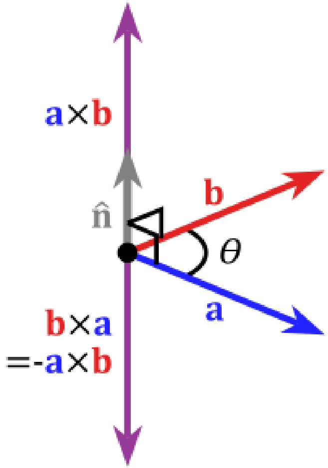
2. Tích có hướng (Cross product):

❖ **Công thức theo hệ tọa độ:**

Với hai vector $\vec{a} = (a_1,\ a_2)$ và $\vec{b} = (b_1,\ b_2)$ thì $a \times b = a_1 b_2 - a_2 b_1 = \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}$

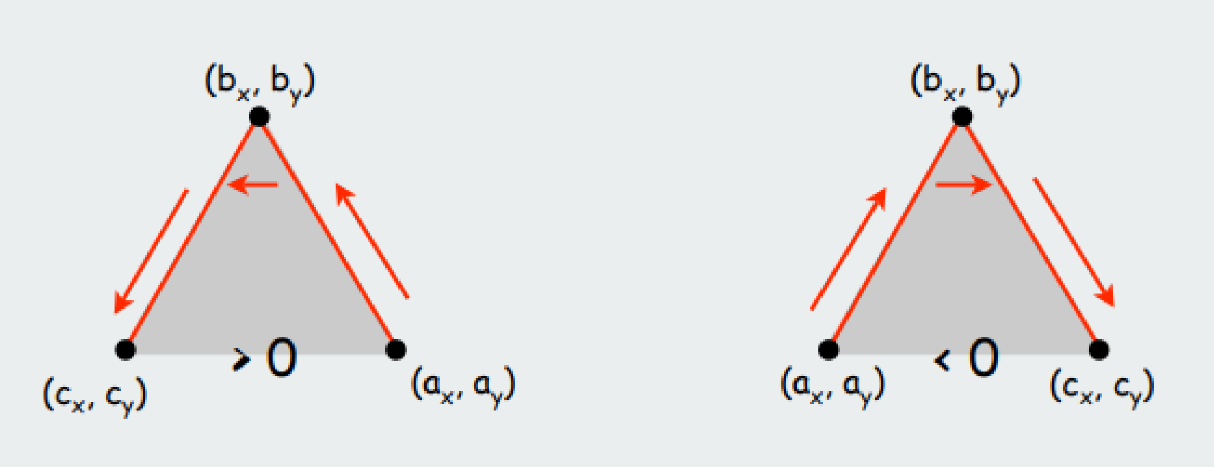
❖ **Về mặt hình học:**

$$\left|\left[\vec{a}, \vec{b}\right]\right| = \left|\vec{a}\right| . \left|\vec{b}\right| . \sin(\vec{a}, \vec{b})$$



B. Tích và góc

Ứng dụng của tích có hướng:



$$2 \times Area(a,b,c) = \begin{vmatrix} a_x & a_y & 1 \\ b_x & b_y & 1 \\ c_x & c_y & 1 \end{vmatrix} = (b_x - a_x)(c_y - a_y) - (b_y - a_y)(c_x - a_x)$$

C. Đường thẳng

1. Định nghĩa và một số tính chất:

- Đường thẳng là tập các điểm (x,y) thuộc phương trình P: a.x + b.y = c *(a^2+ b^2 > 0).*

❖ **Khoảng cách từ một điểm đến đường thẳng:**

- Khoảng cách điểm $A = (a_1, a_2)$ đến đường thẳng P có công thức:

$$dist = \frac{|a.a1+b.a2-c|}{\sqrt{a^2+b^2}}$$

- Bằng cách xét dấu phương trình a.x + b.y - c , ta cũng có thể biết điểm (x,y) nằm phía trái hay phía phải của đường thẳng.

C. Đường thẳng

2. Tìm giao điểm giữa 2 đường thẳng:

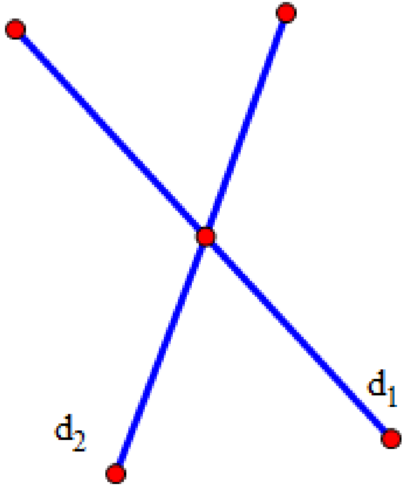
$$\begin{cases} a_1x + b_1y + c_1 = 0 \\ a_2x + b_2y + c_2 = 0 \end{cases}$$

$$m = \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} \qquad n = \begin{vmatrix} b_1 & c_1 \\ b_2 & c_2 \end{vmatrix} \qquad p = \begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix}$$

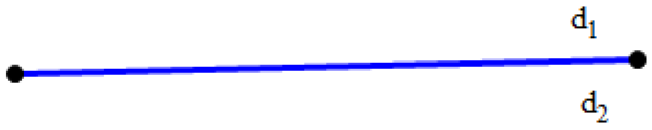
Trường hợp: m ≠ 0

$$x = -\frac{\begin{vmatrix} c_1 & b_1 \\ c_2 & b_2 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}} = -\frac{c_1 b_2 - c_2 b_1}{a_1 b_2 - a_2 b_1},$$

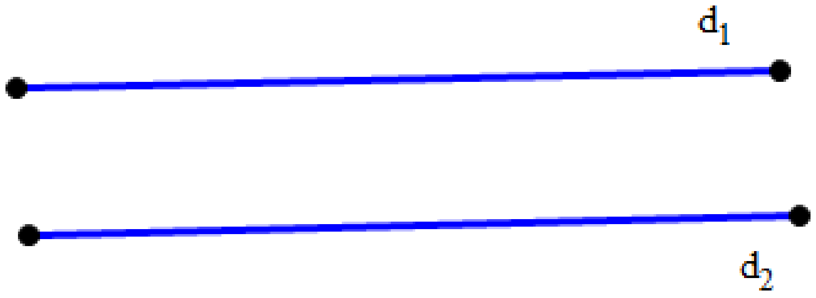
$$y = -\frac{\begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}} = -\frac{a_1 c_2 - a_2 c_1}{a_1 b_2 - a_2 b_1}.$$



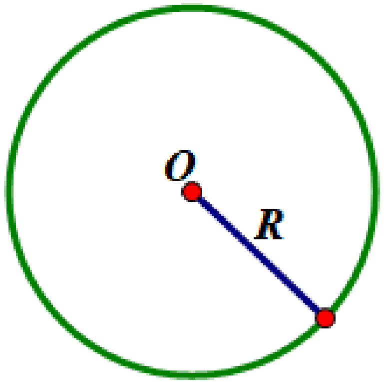
Trường hợp: m = n = p = 0

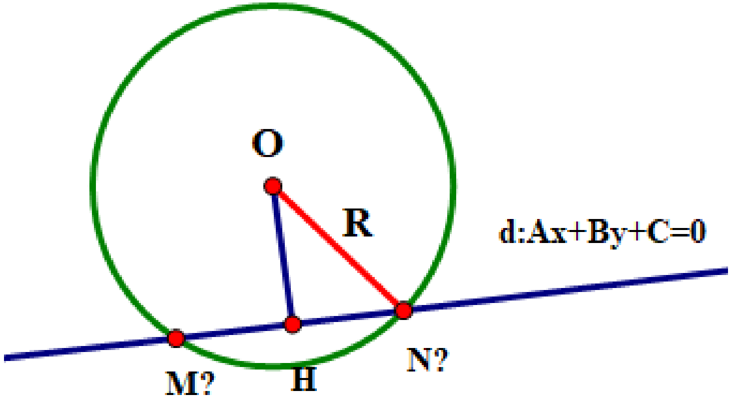


Các trường hợp còn lại:

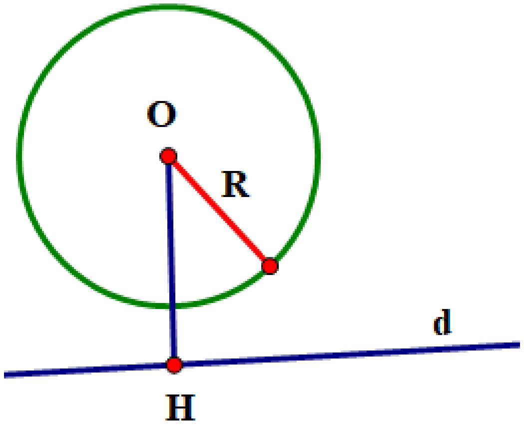


D. ĐƯỜNG TRÒN

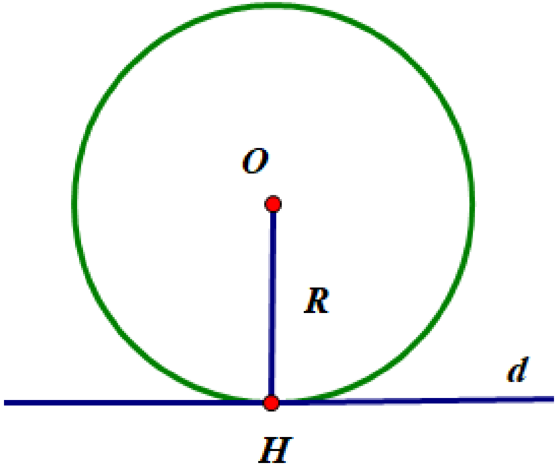




Trường hợp 1:



Trường hợp 2:

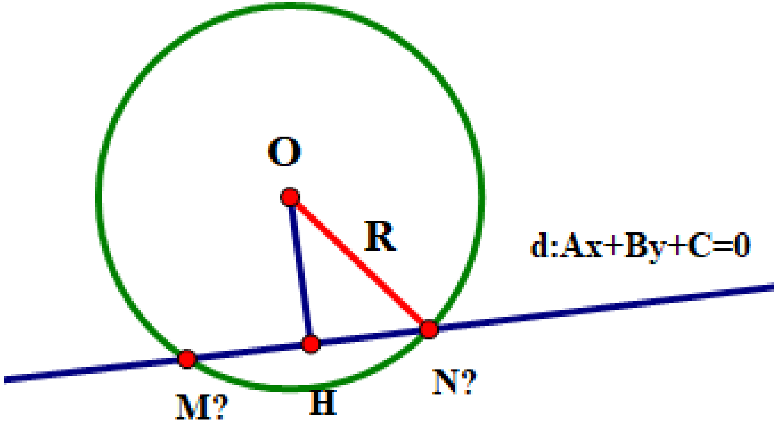


$$d_0 = \frac{|C|}{\sqrt{A^2 + B^2}}$$

$$x_0 = -\frac{AC}{A^2 + B^2}$$

$$y_0 = -\frac{BC}{A^2 + B^2}$$

Trường hợp 3:



$$d_H = \frac{|C|}{\sqrt{A^2 + B^2}} \qquad d = HM = \sqrt{R^2 - d_H{}^2}$$

$$m = \sqrt{\frac{d^2}{A^2 + B^2}}$$

$$\Rightarrow \left[\begin{array}{l} \begin{cases} x_M = x_H + B.m \\ y_M = y_H - A.m \end{cases} \\ \begin{cases} x_N = x_H - B.m \\ y_N = y_H + A.m \end{cases} \end{array}\right.$$

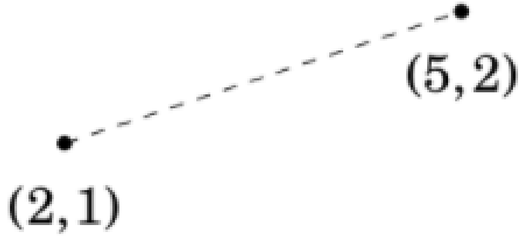
E. Khoảng cách

Hàm khoảng cách định nghĩa khoảng cách giữa các điểm. Hàm khoảng cách thường gặp nhất là **khoảng cách Euclidean** cho biết khoảng cách giữa 2 điểm (x_1, y_1) và (x_2, y_2) là:

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Một hàm khoảng cách khác là **khoảng cách Manhattan** khi khoảng cách giữa 2 điểm (x_1, y_1) và (x_2, y_2) là:

$$|x_1 - x_2| + |y_1 - y_2|$$

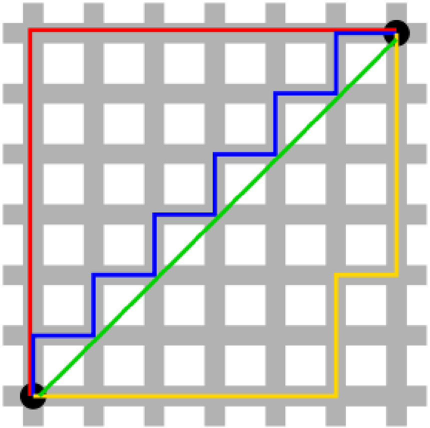


Euclidean distance

$$\sqrt{(5-2)^2+(2-1)^2}=\sqrt{10}$$

Manhattan distance

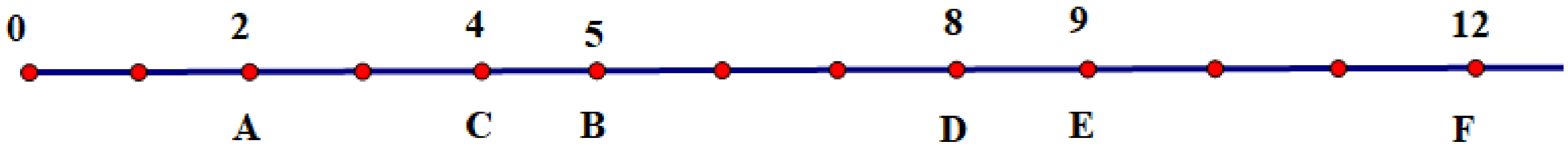
$$|5-2|+|2-1|=4$$



Một vài hàm tính khoảng cách khác

- Hàm Minkowski (p-norm): $$d(x,z)=\left(\sum_{i=1}^{n}|x_i-z_i|^p\right)^{1/p}$$
- Hàm Manhattan ($p=1$): $$d(x,z)=\sum_{i=1}^{n}|x_i-z_i|$$
- Hàm Euclid ($p=2$): $$d(x,z)=\sqrt{\sum_{i=1}^{n}(x_i-z_i)^2}$$
- Hàm Chebyshev ($p=\infty$): $$d(x,z)=\lim_{p\to\infty}\left(\sum_{i=1}^{n}|x_i-z_i|^p\right)^{1/p} = \max_i |x_i-z_i|$$

F. Tổng các phân đoạn



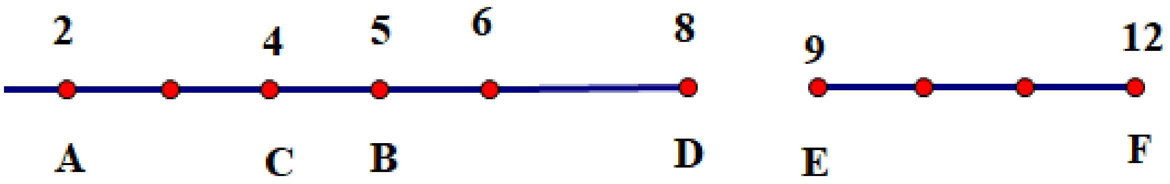
3 segments: AB, CD, EF

Klee's Algorithm

```
for i = 0,      point[0] = {2, start}
                point[1] = {5, end}

for i = 1,      point[2] = {4, start}
                point[3] = {8, end}

for i = 2,      point[4] = {9, start}
                point[5] = {12, end}
```

```
Sorting all points by point value:

points = {2, 5, 4, 8, 9, 12}   →   points = {2, 4, 5, 8, 9, 12}
         {s, e, s, e, s, e}                 {s, s, e, e, s, e}
```

```
points = {2, 4, 5, 8, 9, 12}
         {s, s, e, e, s, e}
```



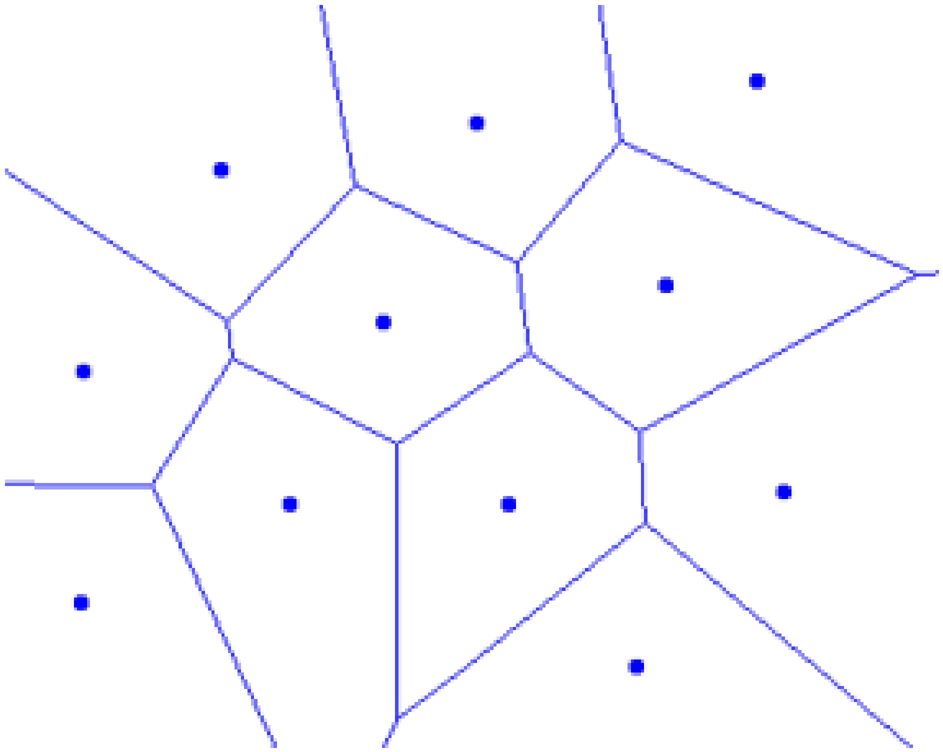

```
for i = 0,    result = 0
              counter = 1

for i = 1,    result = 0 + 4-2
              counter = 1 + 1

for i = 2,    result = 2 + 5-4 = 3
              counter = 2 - 1

for i = 3,    result = 3 + 8-5
              counter = 1 - 1

for i = 4,    result = 6
              counter = 1

for i = 5,    result = 6 + 12-9
              counter = 1 - 1
```

→ Final result = 9

02

SWEEP-LINE

Sweep-line



Sweep-line

Ý tưởng:

- Chúng ta duyệt qua các đường thẳng (sự kiện) từ trái sang phải và duy trì một biến đếm.

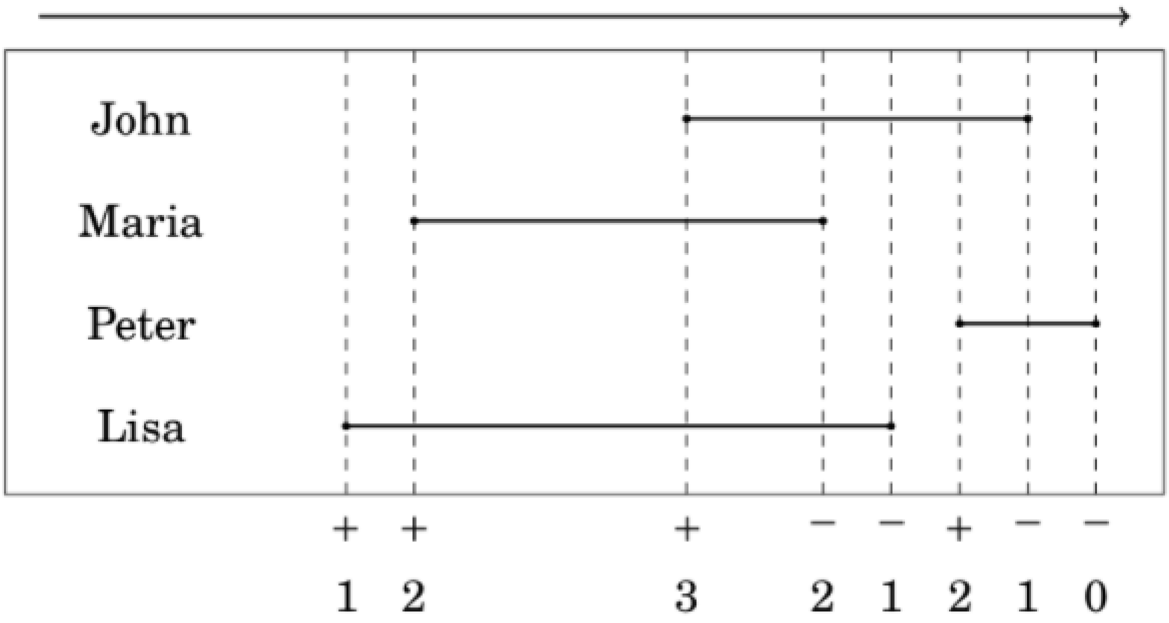
- Khi nào một đường thẳng (sự kiện) đến, chúng ta tăng giá trị của biến đếm lên 1 và khi duyệt xong một đường thẳng, chúng ta giảm biến đếm đi 1.

- Câu trả lời cho bài toán là giá trị max của biến đếm trong suốt thuật toán.

Ví dụ minh họa: Một công ty có **5 nhân viên**, và chúng ta biết rằng mỗi nhân viên có **thời gian đến và về** trong một ngày. Nhiệm vụ của chúng ta là tính **số lượng** nhân viên **lớn nhất có trong văn phòng** tại một thời điểm.

person	arrival time	leaving time
John	10	15
Maria	6	12
Peter	14	16
Lisa	5	13

F. Sweep-line



Thời gian chạy của thuật toán là **O(nlogn),** vì sắp xếp các sự kiện tốn chi phí **O(nlogn),** phần còn lại của thuật toán chỉ mất chi phí O(n).

03

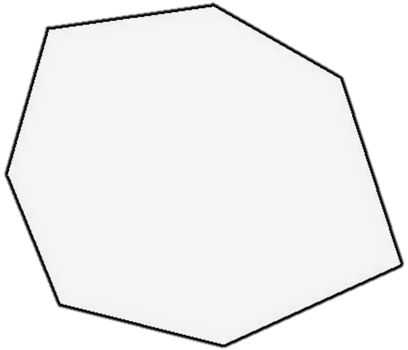
POLYGON

Polygon, what is, Polygon?

A. Đa giác

1. Đa giác:

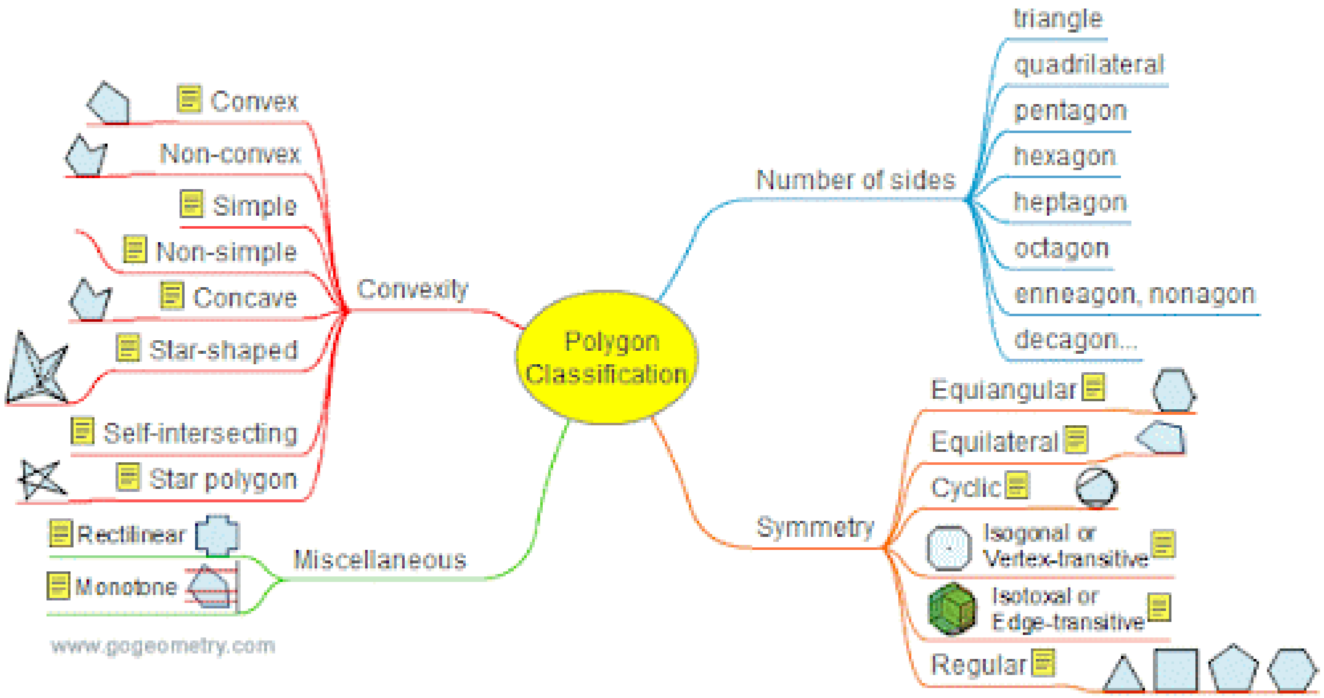
- Trong hình học phẳng, đa giác là một đường gấp khúc phẳng khép kín, nghĩa là gồm những đoạn thẳng nối tiếp nhau, cùng nằm trên một mặt phẳng và khép kín.
- Người ta phân loại đa giác như thế nào và có mấy loại?



A. Đa giác

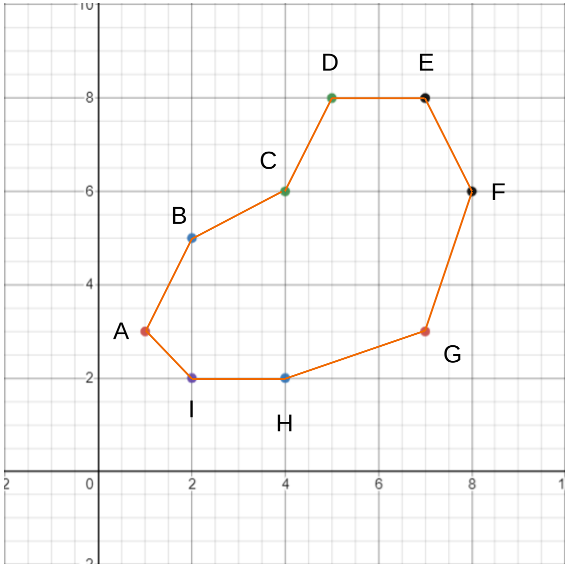
1. Đa giác:

Phân loại các đa giác:



A. Đa giác

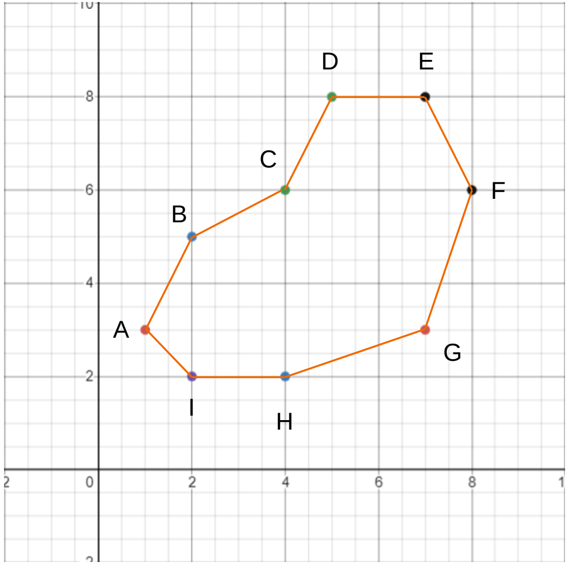
2. Diện tích của một đa giác:

Area of  = ?

A. Đa giác

2. Diện tích của một đa giác:

Area of

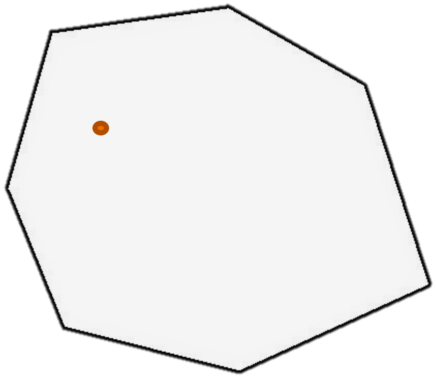


$$= \sum_{(p,q)\in \text{edges}} \frac{(p_x - q_x)\cdot(p_y + q_y)}{2}$$

A. Đa giác

3. Kiểm tra một điểm nằm trong đa giác:

Viết một đoạn code bằng python (hoặc mã giả). Kiểm tra xem một điểm có nằm trong đa giác hay không?



A. Đa giác

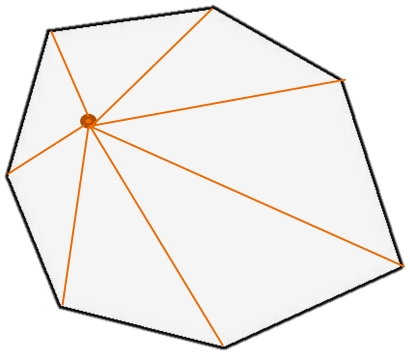
3. Kiểm tra một điểm nằm trong đa giác:

Dành cho đa giác lồi.

- Chia đa giác thành các tam giác nhỏ với mỗi tam giác là 2 đỉnh của một cạnh và đỉnh còn lại chính là điểm ta cần kiểm tra.

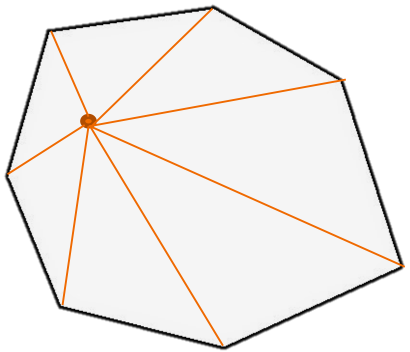
- Tính tổng diện tích của các tam giác, nếu tổng diện tích của chúng bằng với diện tích đa giác thì điểm đó sẽ nằm trong đa giác.

=> Độ phức tạp: $\boldsymbol{O(n)}$



A. Đa giác

```
def checkPointInConvex(PointsList, Point):
    TriangleSum = 0.0
    for i in range(len(PointsList)):
        TriangleSum += getTriangleArea(Points[i - 1], Point, Points[i])
    return (TriangleSum == getPolygonArea(PointsList))
```



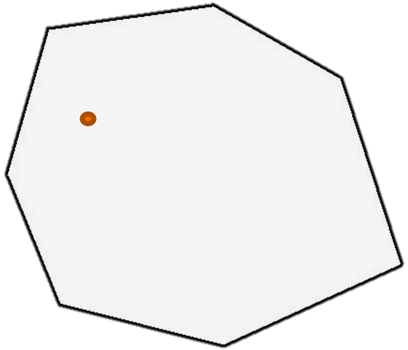
A. Đa giác

3. Kiểm tra một điểm nằm trong đa giác:

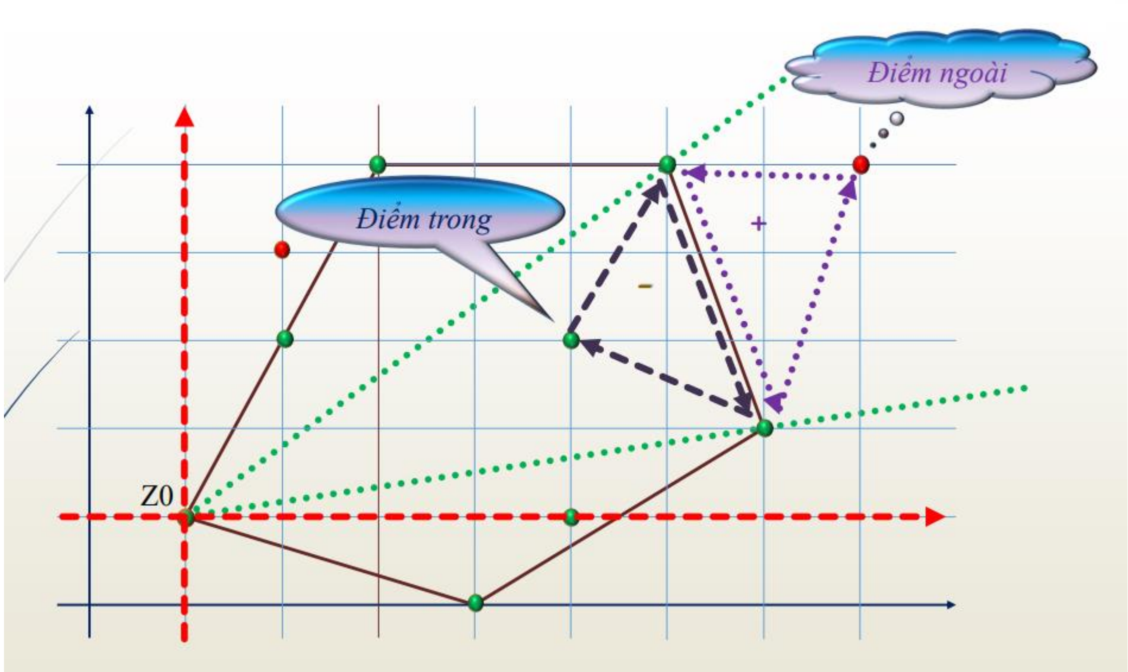
Dành cho đa giác lồi.

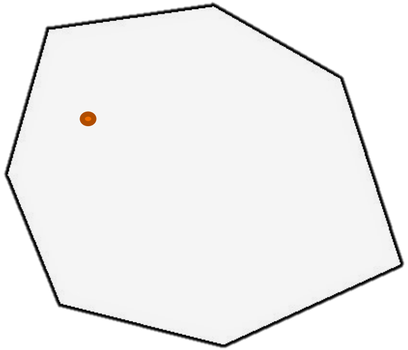
- Xác định điểm trái dưới của đa giác (trái nhất của thấp nhất).
- Tìm kiếm nhị phân điểm i sao cho điểm i và điểm liền kề nó tạo thành 1 góc tọa độ cực kẹp góc tọa độ cực được tạo bởi điểm cần xét.
- Nếu 3 điểm i, i + 1 và điểm cần xét có thứ tự theo chiều kim đồng hồ thì điểm cần xét nằm trong đa giác, ngược lại thì không.

=> Độ phức tạp: $\boldsymbol{O(log_2 n)}$



A. Đa giác

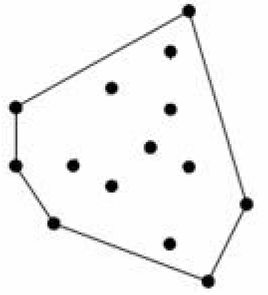


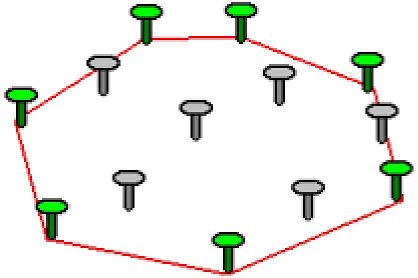


B. Bao Lồi

1. Bao Lồi:

- Trong **hình học tính toán** (**computational geometry**), **bao lồi** (**convex hull**) của một tập điểm là tập lồi nhỏ nhất (theo diện tích, thể tích, …) mà tất cả các điểm đều nằm trong tập đó.
- Hình ảnh bao lồi có thể dễ dàng nhìn thấy ở xung quanh.

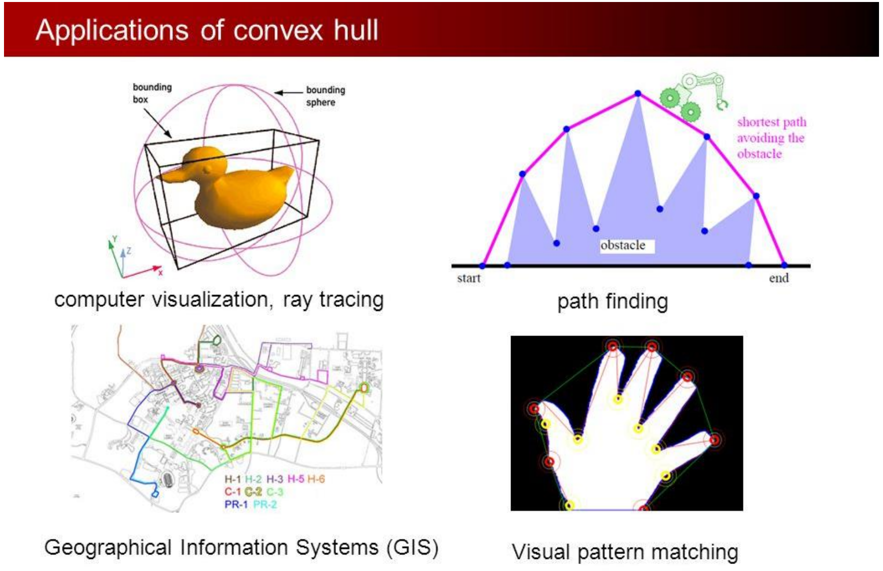




B. Bao Lồi

2. Ứng dụng của bao lồi:

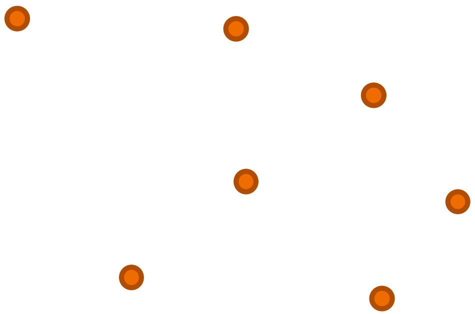
Chính vì những đặc điểm của bao lồi khiến nó có vai trò cực kì quan trọng trong nhiều lĩnh vực.



B. Bao Lồi

3. Thuật toán xác định bao lồi:

Vì bao lồi quan trọng đến thế, vì vậy, cho trước một tập hợp các điểm trên mặt phẳng, làm sao để tìm bao lồi của chúng?

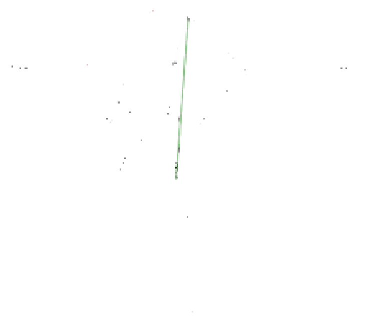


B. Bao Lồi

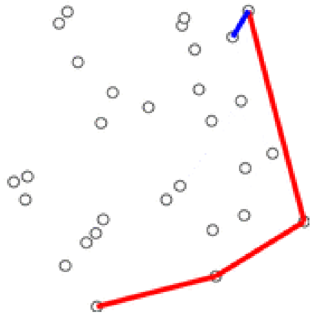
3. Thuật toán xác định bao lồi:

Các thuật toán xác định bao lồi phổ biến:

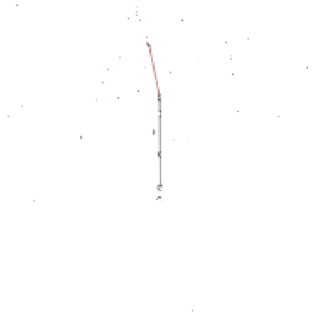
- Thuật toán bọc gói (Gift Wrapping) – Độ phức tạp (trong TH xấu nhất): $O(n^2)$
- Thuật toán Graham – Độ phức tạp: $O(nlogn)$
- Thuật toán chuỗi đơn điệu (Monotone Chain) – Độ phức tạp: $O(nlogn)$



Thuật toán bọc gói



Thuật toán Graham



Thuật toán Monotone Chain

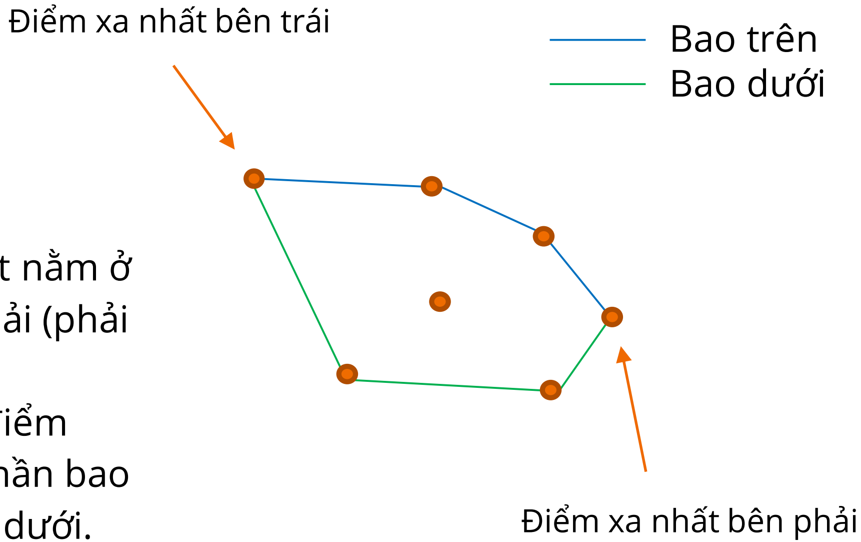
B. Bao Lồi

3. Thuật toán xác định bao lồi:

Thuật toán Chuỗi đơn điệu (Monotone chain)

Ý tưởng:

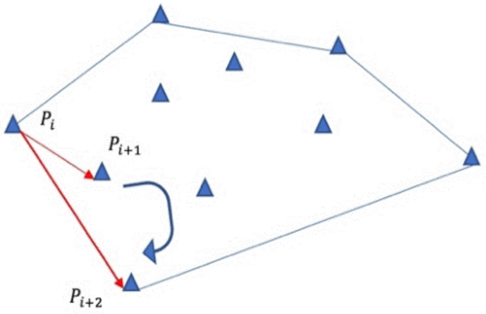
- Sắp xếp các điểm lại theo hoành độ, điểm xa nhất nằm ở bên trái (trái nhất) và điểm xa nhất nằm ở bên phải (phải nhất) luôn luôn là hai đỉnh của bao lồi.
- Phần bao lồi từ ở phía trên từ điểm trái nhất tới điểm phải nhất (theo chiều kim đồng hồ) được gọi là phần bao trên. Ngược lại, phần ở phía dưới được gọi là bao dưới.



B. Bao Lồi

3. Thuật toán xác định bao lồi:

Thuật toán Chuỗi đơn điệu (Monotone chain)



Thuật toán:

- Sắp xếp các điểm lại theo hoành độ, nếu cùng hoành độ thì điểm có tung độ bé hơn sẽ đứng trước.
- Gọi U là tập hợp các điểm nằm ở bao trên và u là số lượng. Xét tại một điểm.
- ❑ Thêm điểm này vào tập U, nếu tập U vẫn ít hơn 3 phần tử thì tiếp tục thêm điểm tiếp theo.
- ❑ Xét $\vec{a} = \overrightarrow{U_{u-2}U_{u-1}}$ và $\vec{b} = \overrightarrow{U_{u-1}U_u}$. Nếu $\vec{a} \times \vec{b} < 0$ thì ta xét tiếp điểm tiếp theo, còn không, loại bỏ điểm U_{u-1} và quay lại bước đầu tiên.
- Thực hiện tương tự với bao dưới, theo thứ tự ngược lại, ta sẽ tìm được bao lồi.

B. Bao Lồi

3. Thuật toán xác định bao lồi:

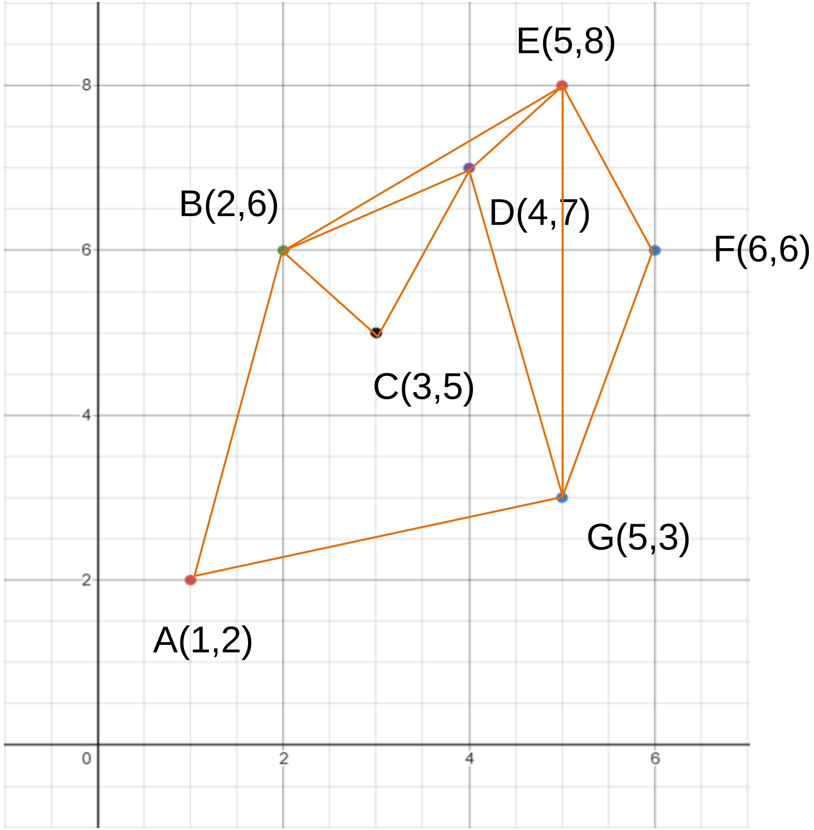
Thuật toán Chuỗi đơn điệu (Monotone chain)

- Bao trên: theo chiều kim đồng hồ.
  - $\overrightarrow{AB} \times \overrightarrow{BC} = -5 < 0$
  - $\overrightarrow{BC} \times \overrightarrow{CD} = 3 > 0$
  - $\overrightarrow{BD} \times \overrightarrow{DG} = -9 < 0$
  - $\overrightarrow{DG} \times \overrightarrow{GE} = 5 > 0$
  - $\overrightarrow{BD} \times \overrightarrow{DE} = 1 > 0$
  - $\overrightarrow{BE} \times \overrightarrow{EF} = -8 < 0$

=> Bao trên gồm: A, B, E, F

- Bao dưới: ngược chiều kim đồng hồ.
  - $\overrightarrow{AG} \times \overrightarrow{GF} = 11 > 0$

=> Bao dưới gồm: A, G, F



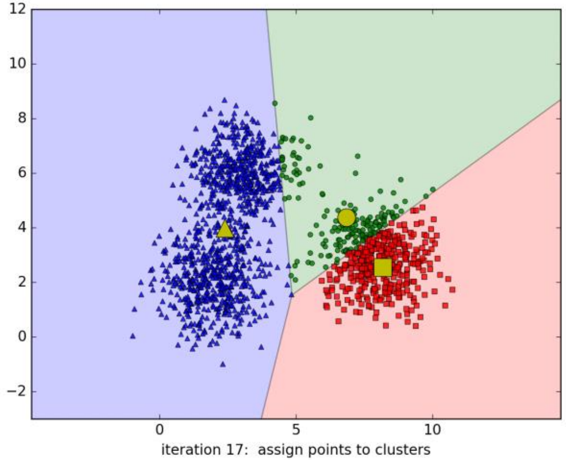
04

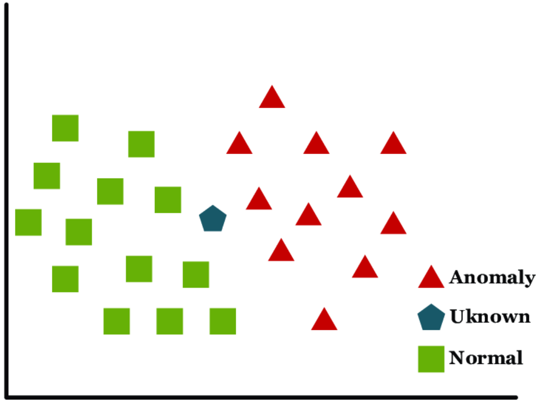
APPLICATIONS

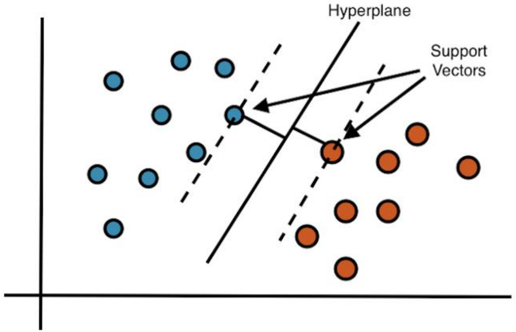
Applications of Geometry

Applications

1. Machine Learning:



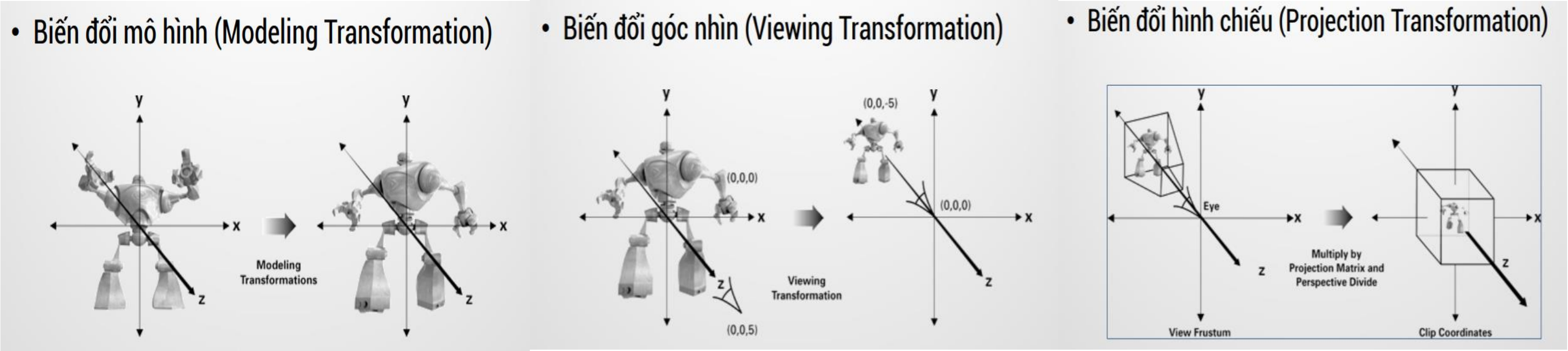




K-mean Clustering **K-nearest Neighbor** **Support Vector Machine**

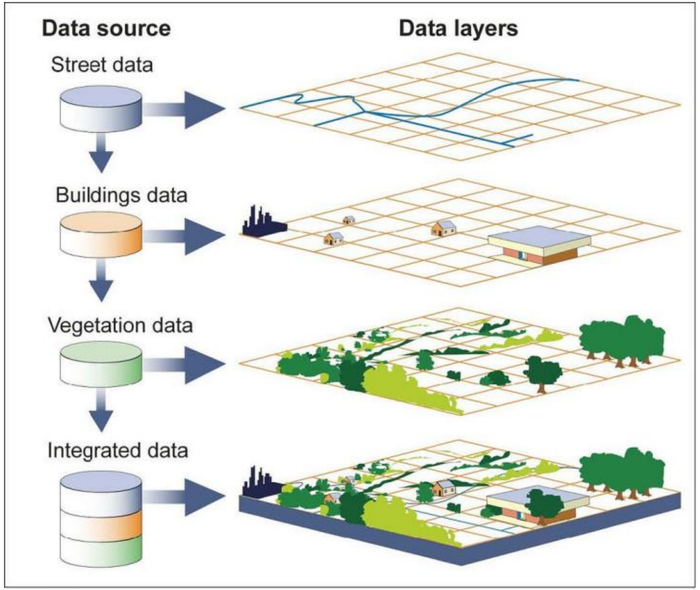
Applications

2. Thuật toán Affine trong đồ họa máy tính:

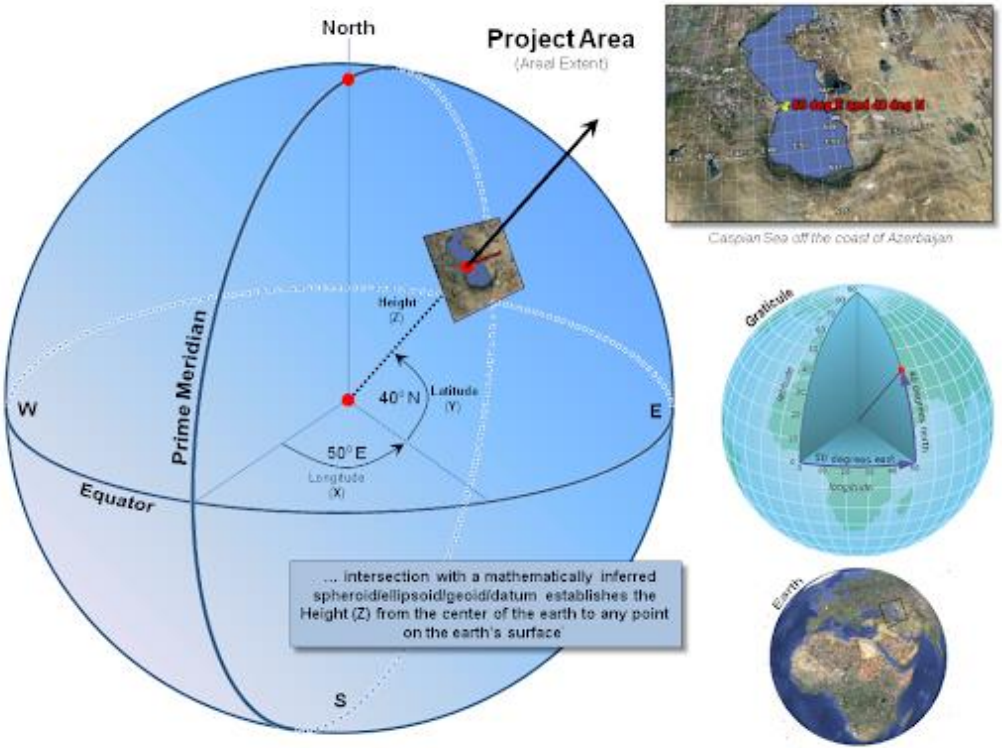


Applications

3. Hệ thống thông tin địa lý:



Source: GAO.



THANKS FOR YOUR PARTICIPATION ❤❤