

## 4.1 Convex hull

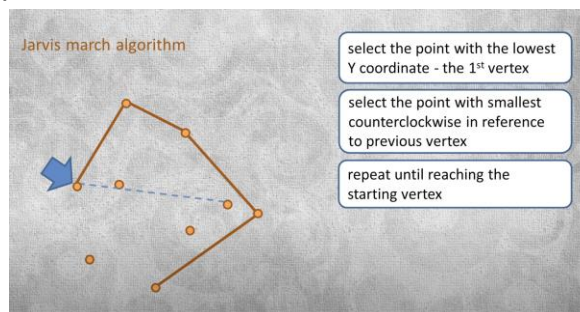
**Victor Misael Escalante Alvarado, A01741176**

## Explicación del algoritmo elegido : Jarvis march

El algoritmo Jarvis March, también conocido como el algoritmo de envoltura (según YouTube “wrapping”), es una técnica utilizada para encontrar el casco convexo de un conjunto de puntos en un plano. Este algoritmo funciona seleccionando un punto inicial y luego envolviendo los puntos uno por uno, siempre eligiendo el siguiente punto que forma el ángulo más pequeño con la línea actual. Este proceso se repite hasta que se vuelve al punto inicial, formando así el casco convexo. La principal característica del algoritmo Jarvis March es su simplicidad y su capacidad para manejar casos con un número pequeño de puntos de manera eficiente.

Los pasos del algoritmo Jarvis March son los siguientes:

1. Seleccionar el punto más bajo (lowest y coordinate).
2. Desde el punto inicial, buscar el punto que forme el ángulo más pequeño (y en sentido antihorario).
3. Agregar este punto al casco convexo.
4. Repetir el proceso desde el nuevo punto y continuar hasta que el siguiente punto sea el punto inicial



Mientras que Jarvis March tiene una complejidad de  $O(n^2)$ , el algoritmo de Graham tiene una complejidad de  $O(n \log n)$ , lo que lo hace más adecuado para conjuntos de puntos más grandes.

## Capturas del código funcionando:

Codigo :

### Algoritmo de jarvis

```
# encuentra el punto mas a la izquierda para ser el inicial
def Left_index(points):

    minn = 0
    for i in range(1, len(points)):
        if points[i][0] < points[minn][0]:
            minn = i
        elif points[i][0] == points[minn][0]:
            if points[i][1] > points[minn][1]:
                minn = i
    return minn

def convexHull(points, n):

    # Encontrar el punto mas a la izquierda
    l = Left_index(points)
    hull = []

    p = l
    q = 0
    while(True):
        hull.append(p)
        q = (p + 1) % n

        for i in range(n):
            if orientacion(points[p], points[i], points[q]) == -1:
                q = i

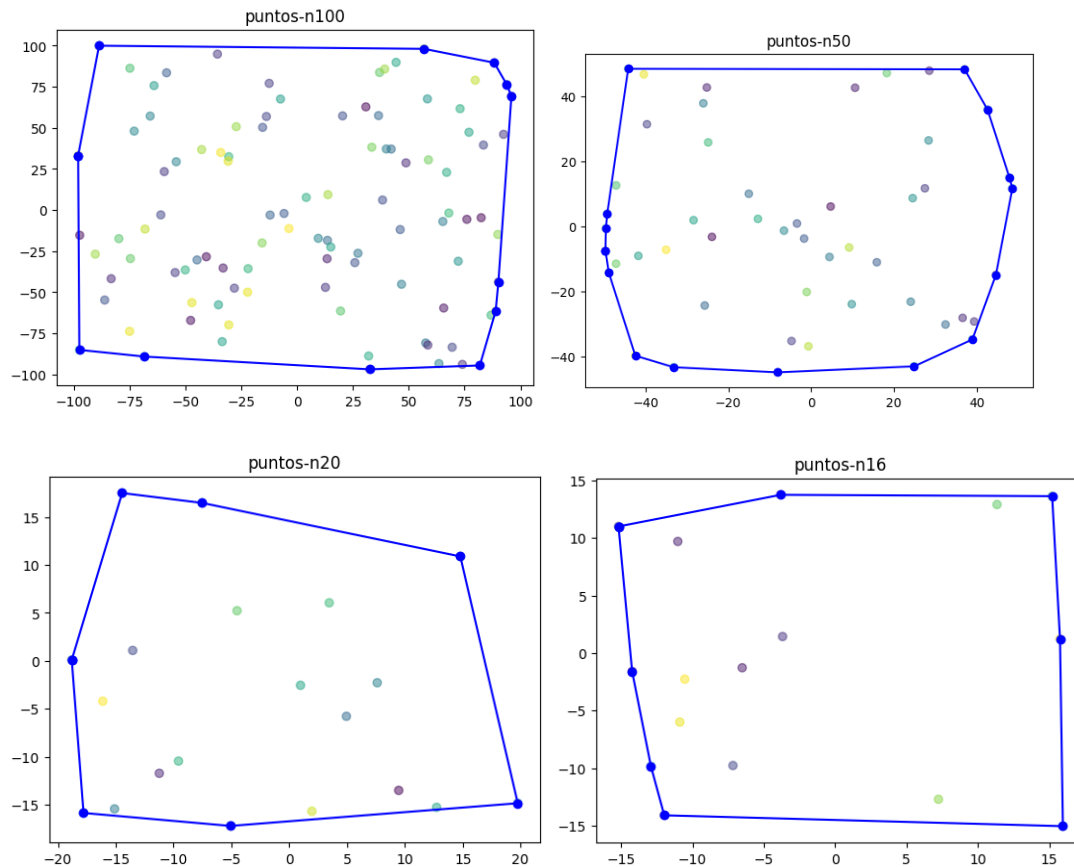
        p = q

        if p == l:
            break

    elegidos = [points[i] for i in hull]
    return elegidos, l
```

✓ 0.0s

## Respuestas :



## Enlace a colab:

[https://drive.google.com/file/d/1YX\\_0qfYD73ce35KiKGLldyHCr0Huz\\_Ps/view?usp=sharing](https://drive.google.com/file/d/1YX_0qfYD73ce35KiKGLldyHCr0Huz_Ps/view?usp=sharing)

## Referencias:

(2024). Youtube.com. <https://www.youtube.com/watch?v=B2AJoSZf4M>

*Casco convexo / Conjunto 1 (Algoritmo de Jarvis o Wrapping) – Barcelona Geeks.* (2022, July 5). Barcelonageeks.com. <https://barcelonageeks.com/casco-convexo-conjunto-1-algoritmo-o-envoltura-de-jarvis/>