

# CONVEX HULL IN 2D

Vera Sacristán

Discrete and Algorithmic Geometry  
Geometria  
Facultat de Matemàtiques i Estadística  
Universitat Politècnica de Catalunya

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

--

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme points

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
-- --  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $n_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

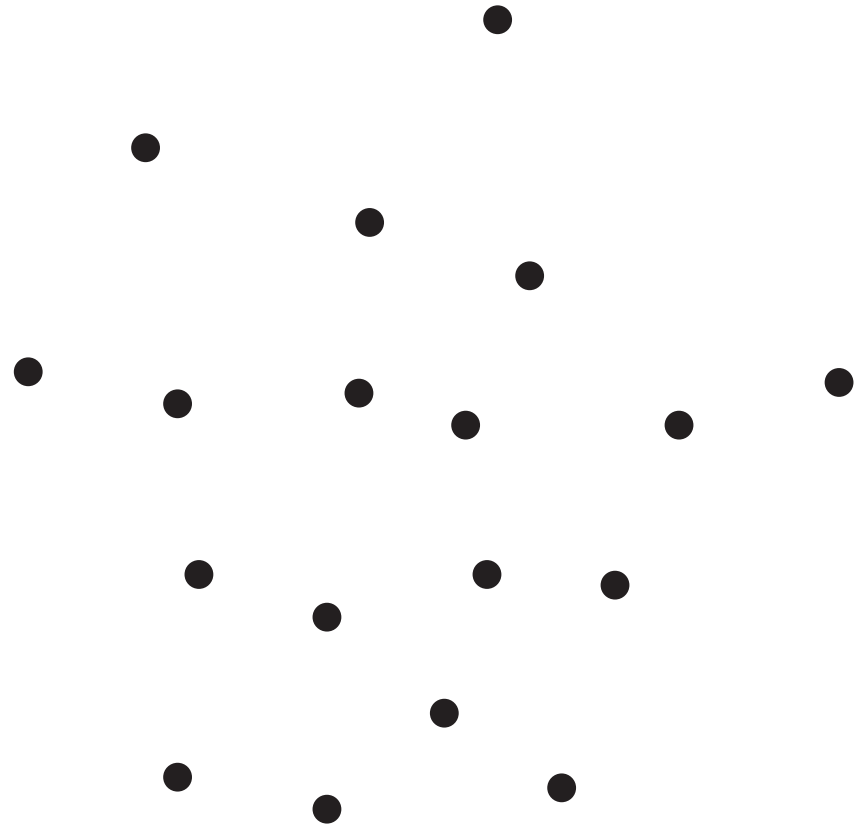
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



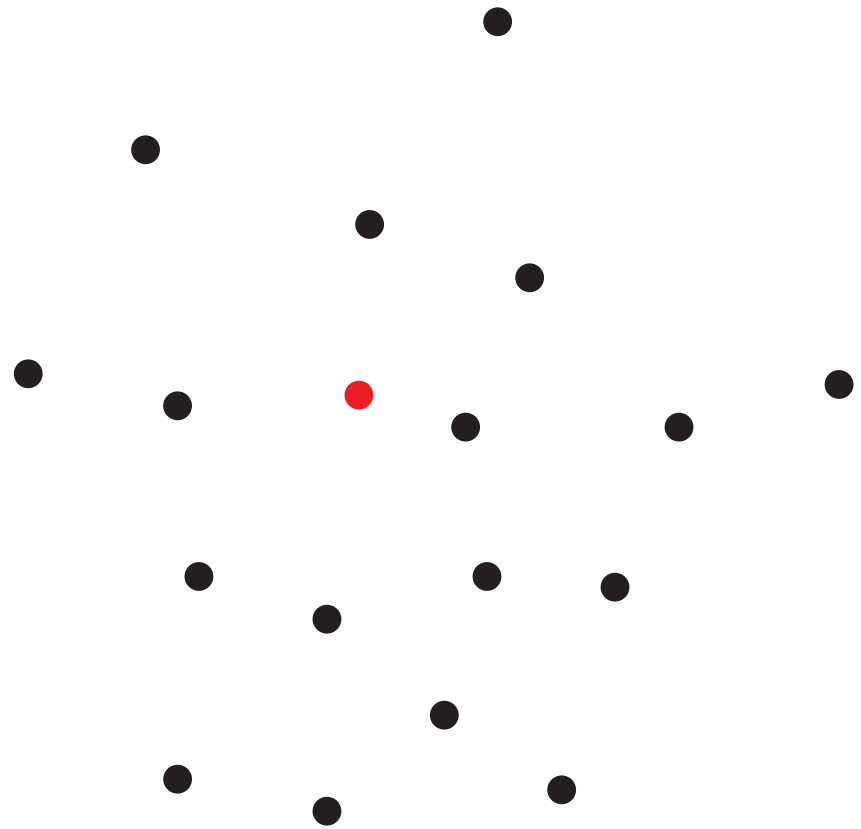
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
-- --  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



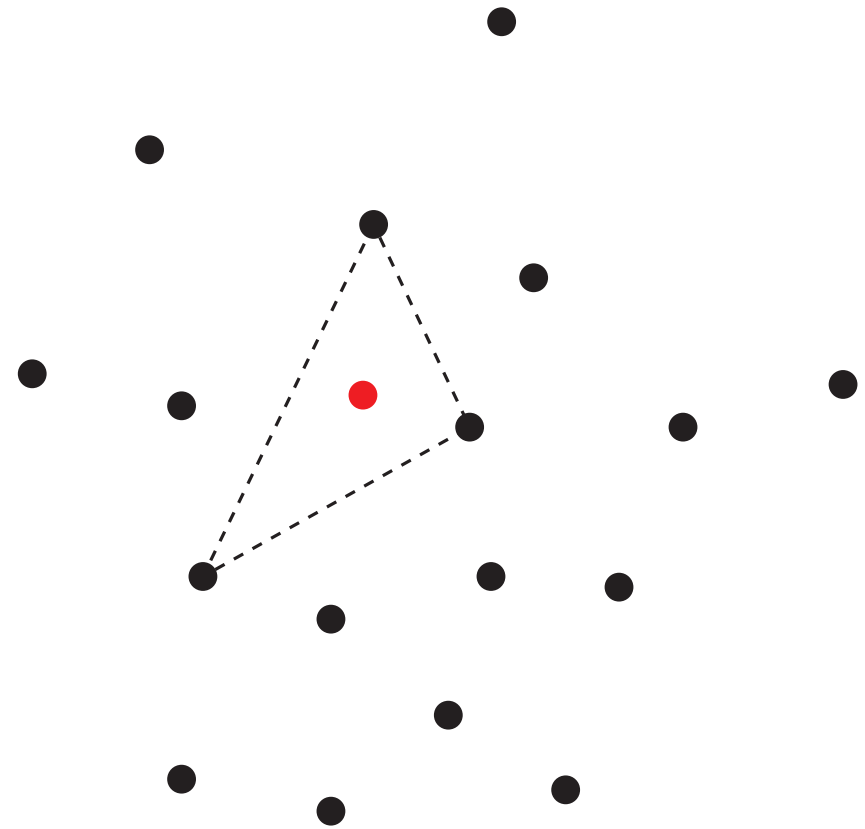
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

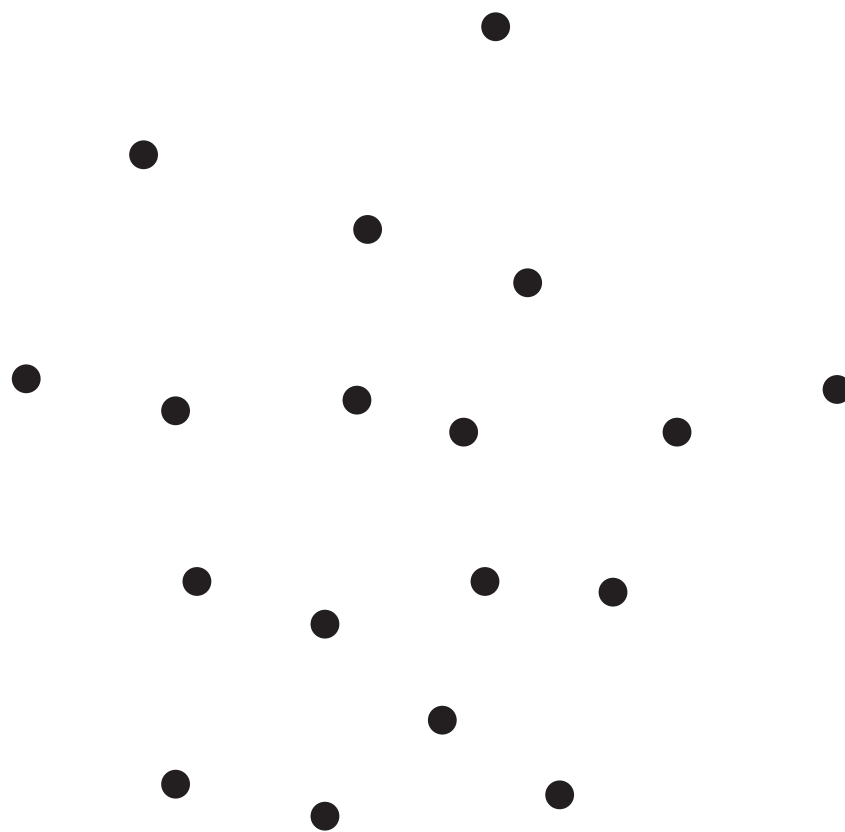
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $n_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

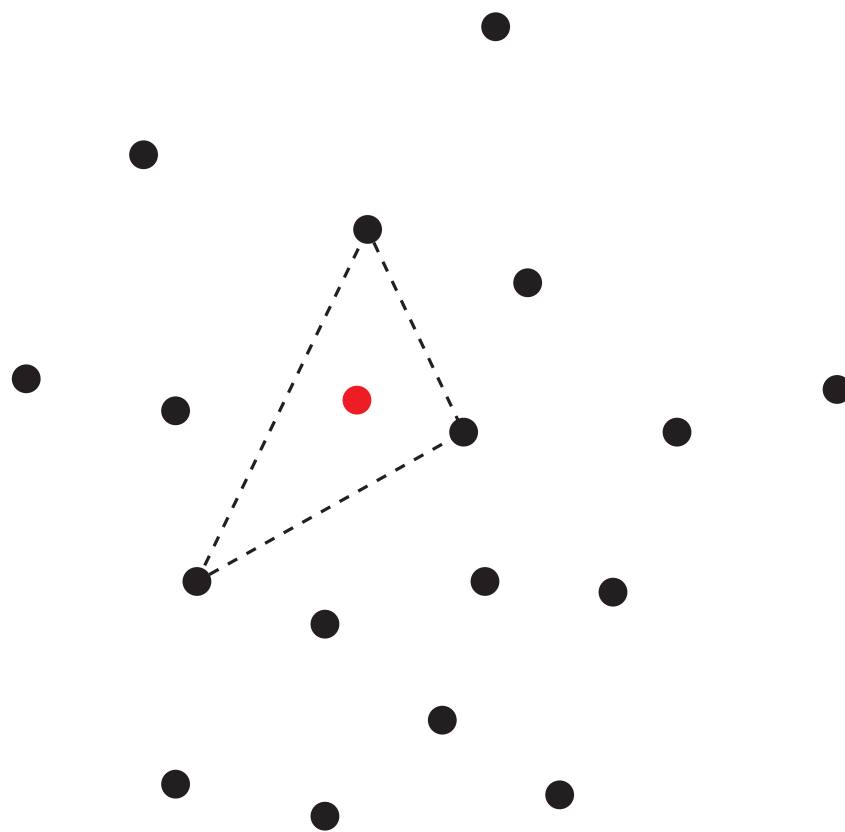
Algo  
Input  
Output  
Process  
Flow  
R

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until all remaining points are extreme points.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Process  
Flow  
R



# CONVEX HULL IN 2D

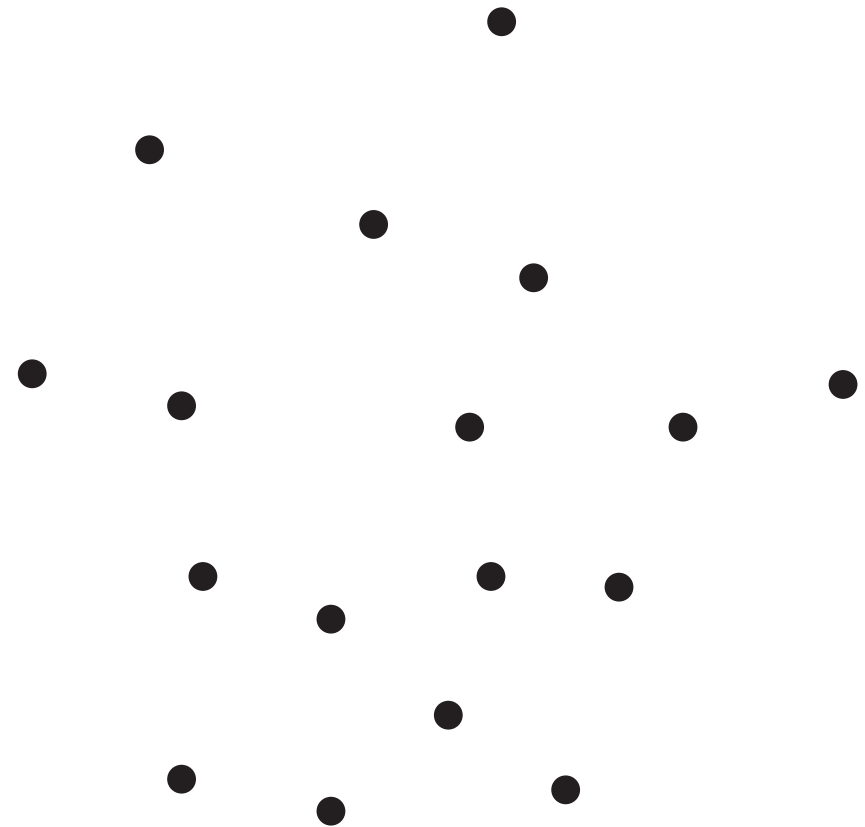
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

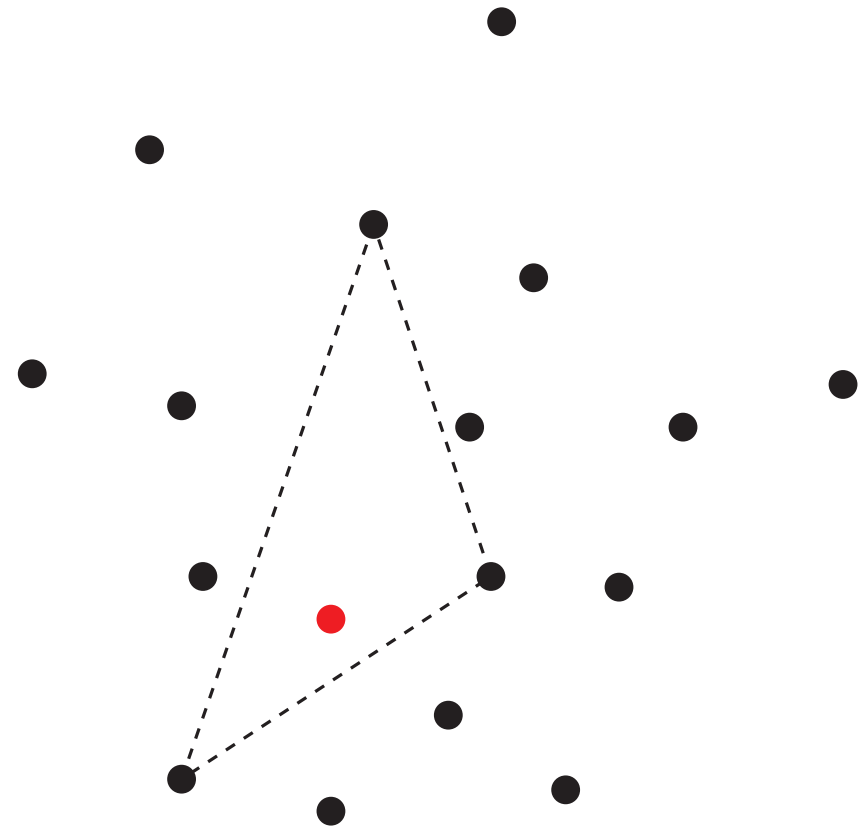
Algo  
Input  
Output  
Process  
Flow  
R

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until all remaining points are extreme points.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Process  
Flow  
R

# CONVEX HULL IN 2D

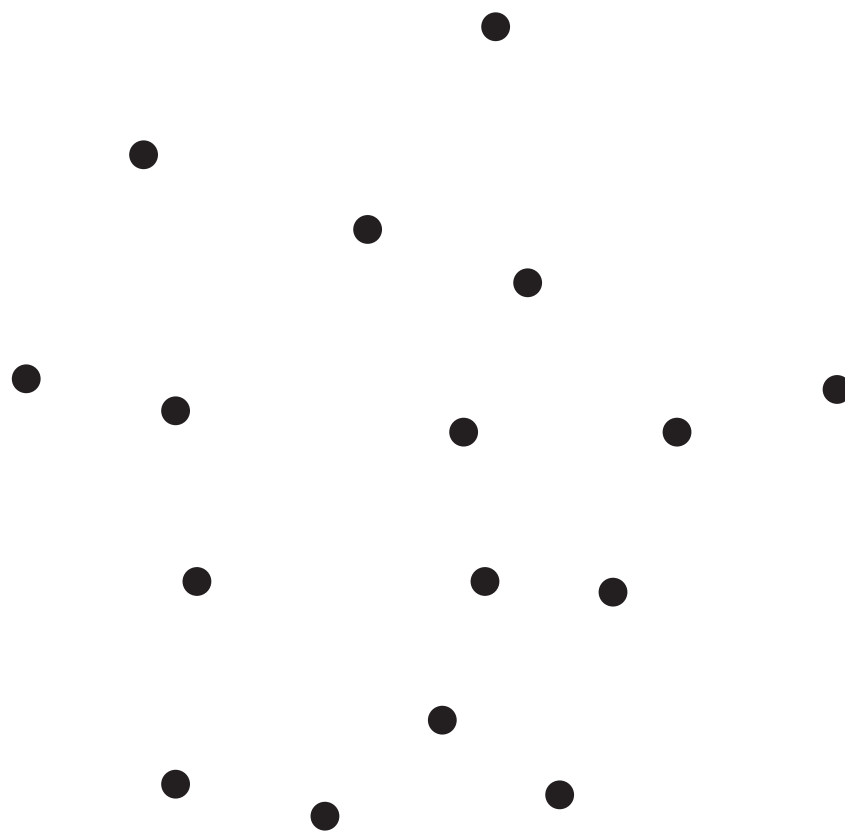
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Algo  
Input  
Output  
Process  
Flow  
R

# CONVEX HULL IN 2D

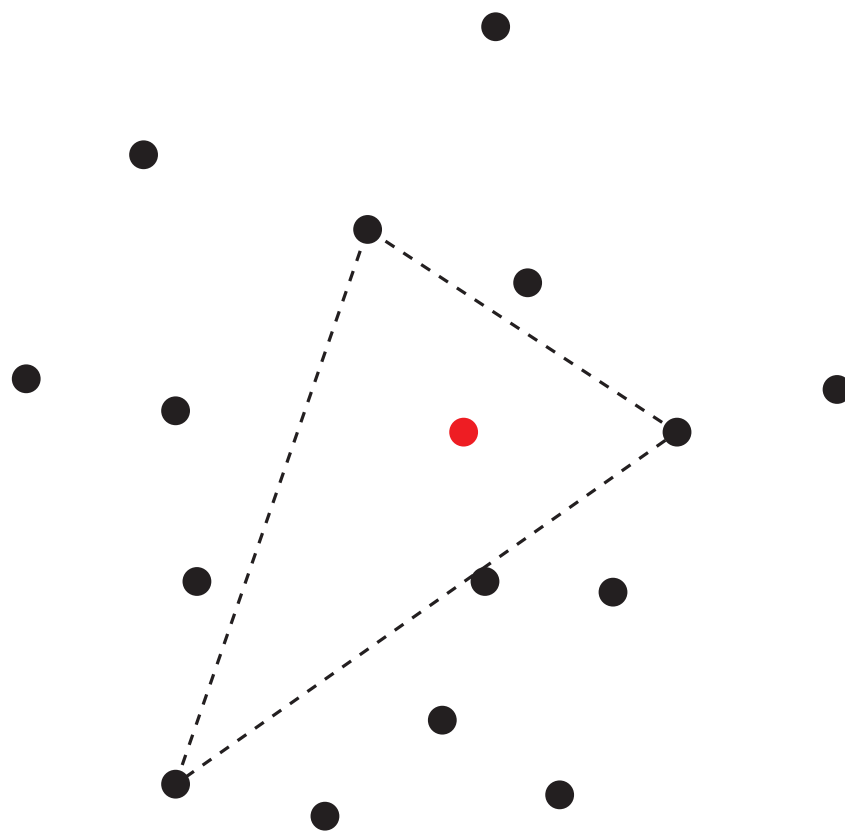
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

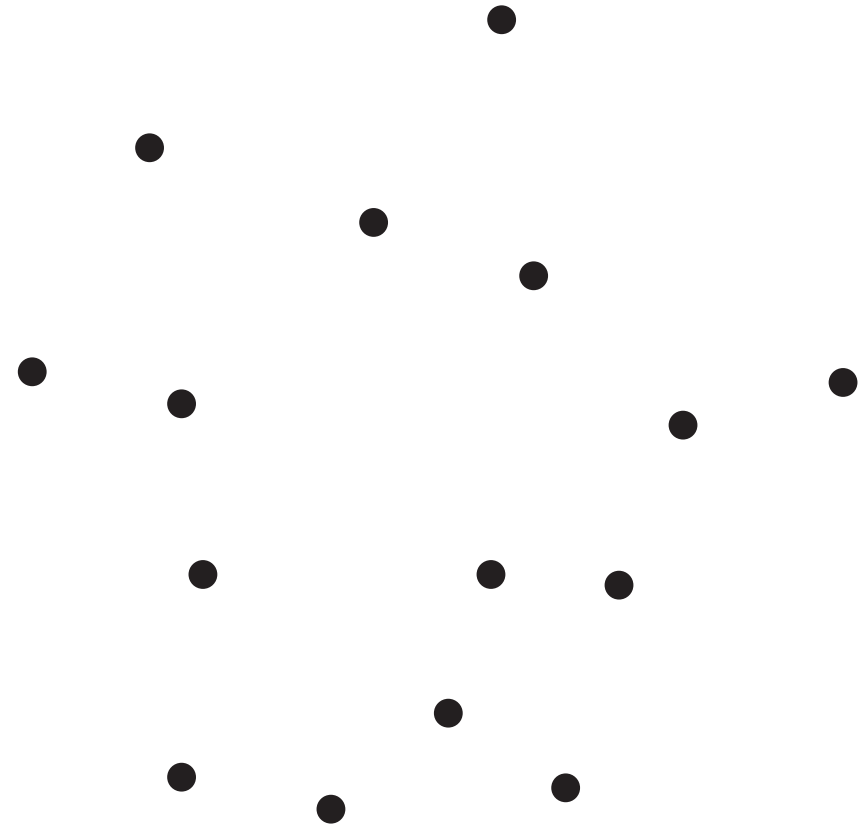
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

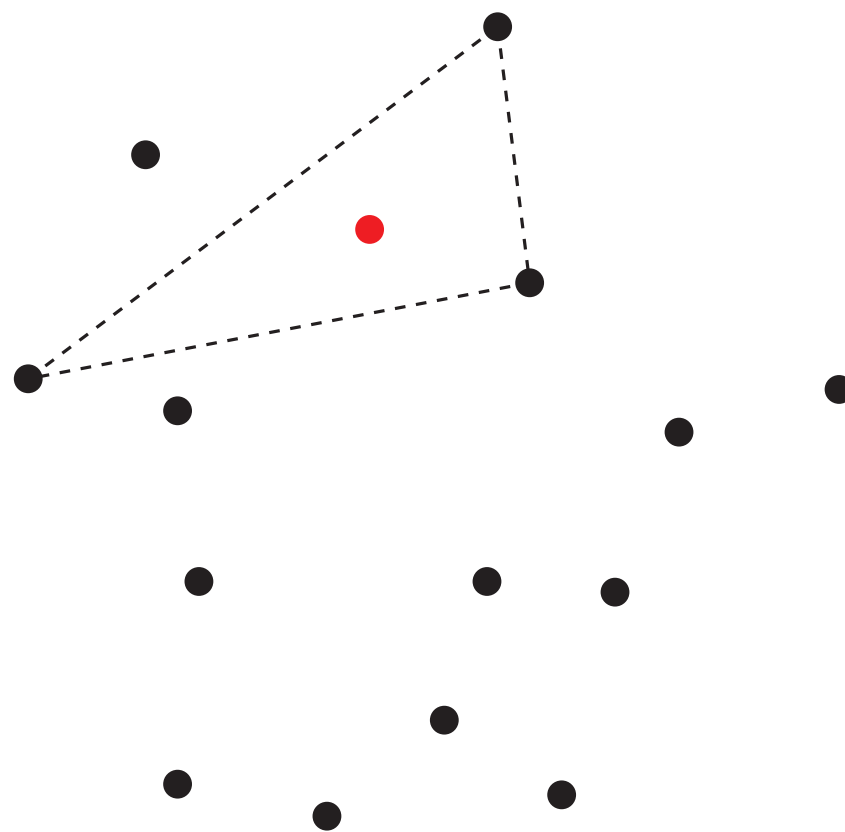
Algo  
Input  
Output  
Process  
Flow  
R

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Process  
Flow  
R

# CONVEX HULL IN 2D

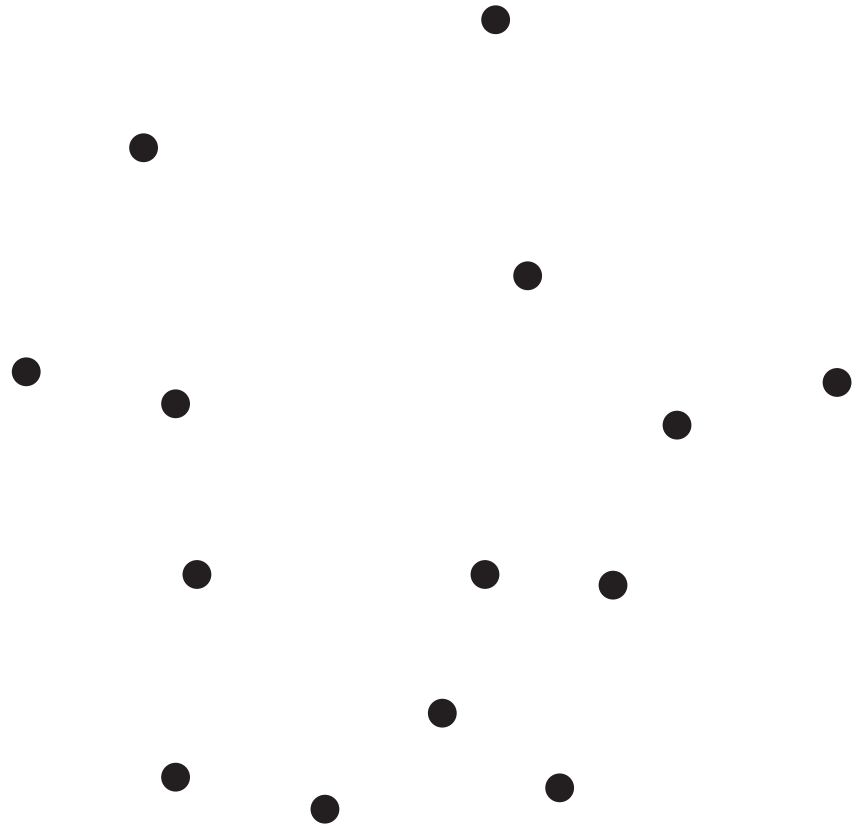
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

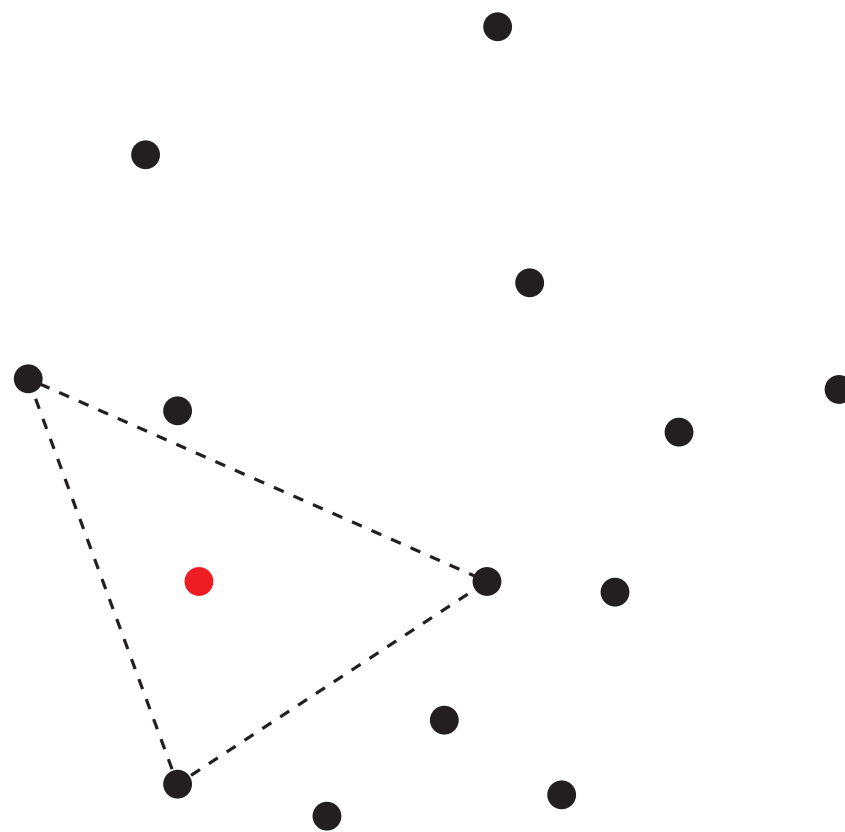
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Proc  
To  
R



# CONVEX HULL IN 2D

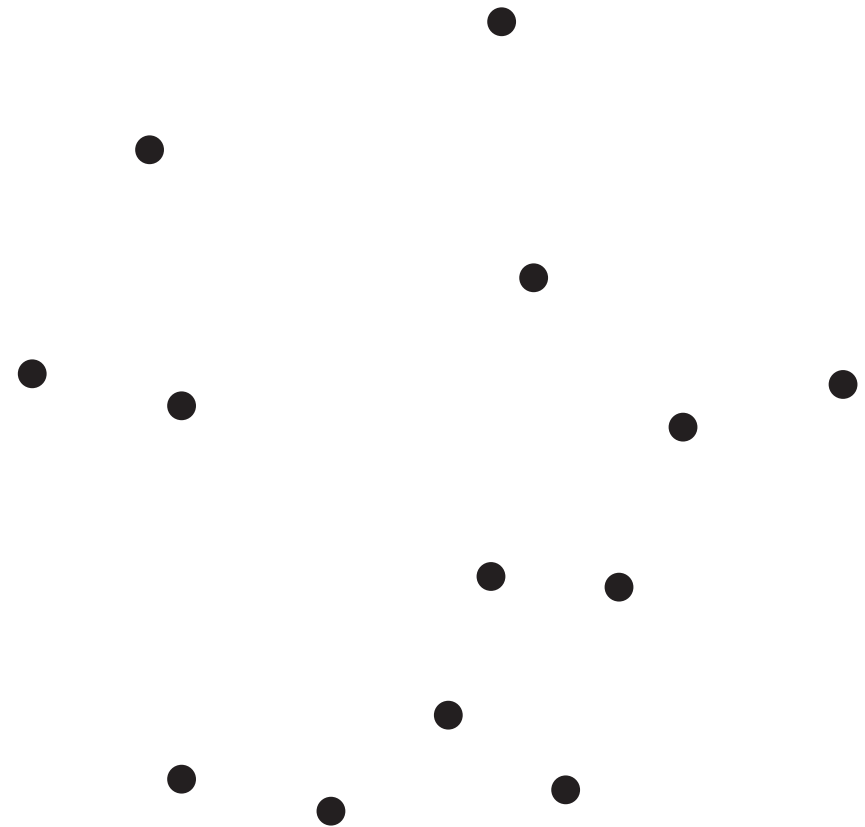
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

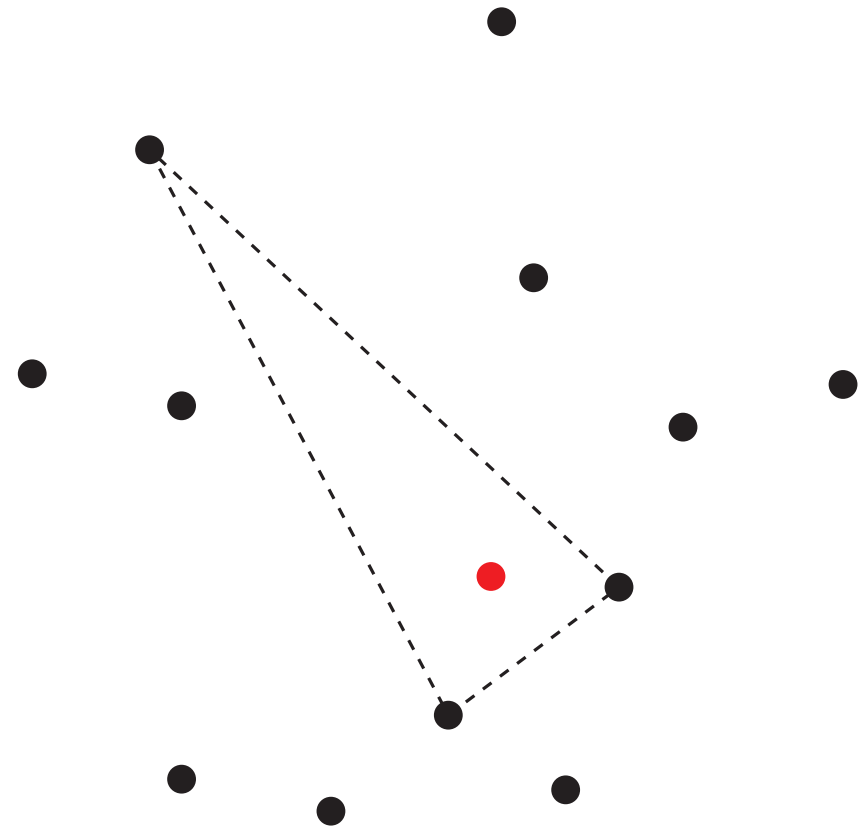
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

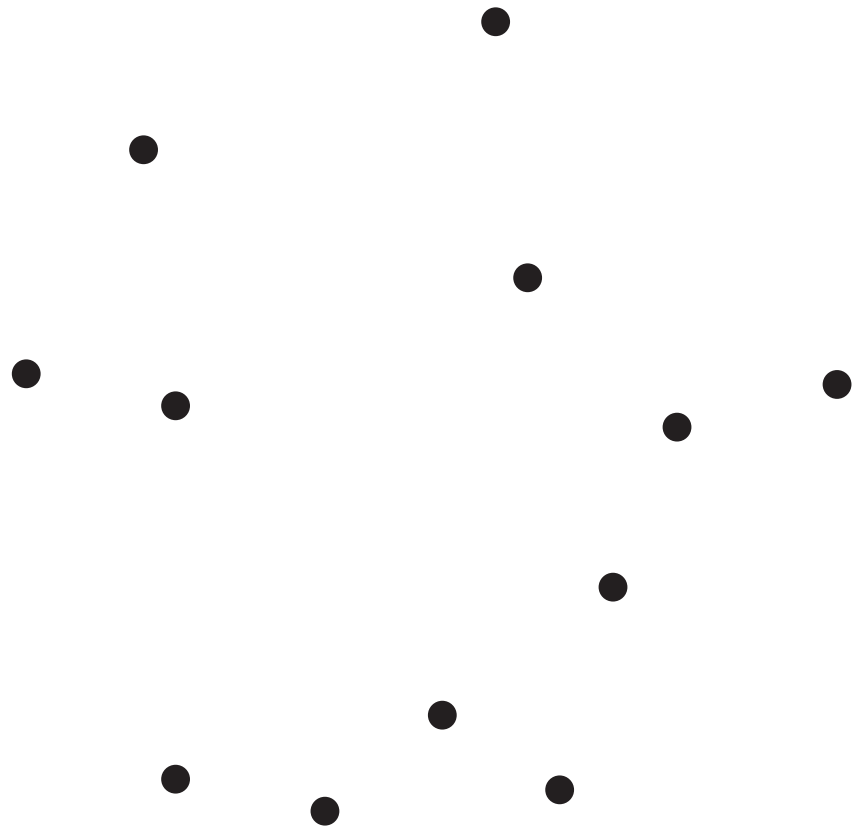
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

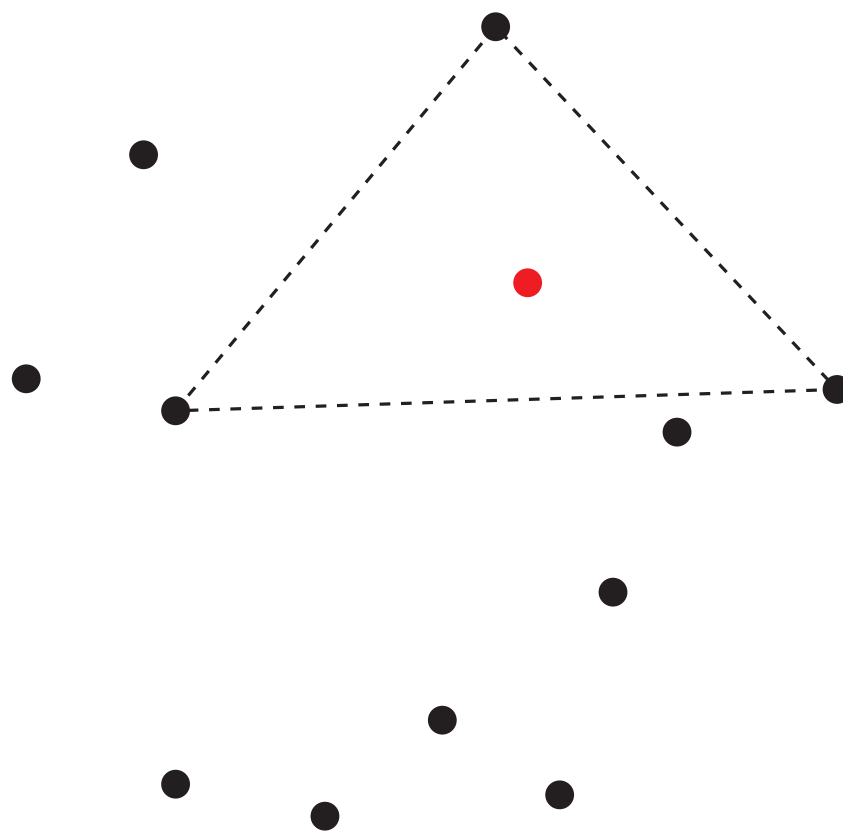
Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until all remaining points are surviving  $p_i$ 's.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Algo  
Input  
Output  
Process  
Flow  
R

# CONVEX HULL IN 2D

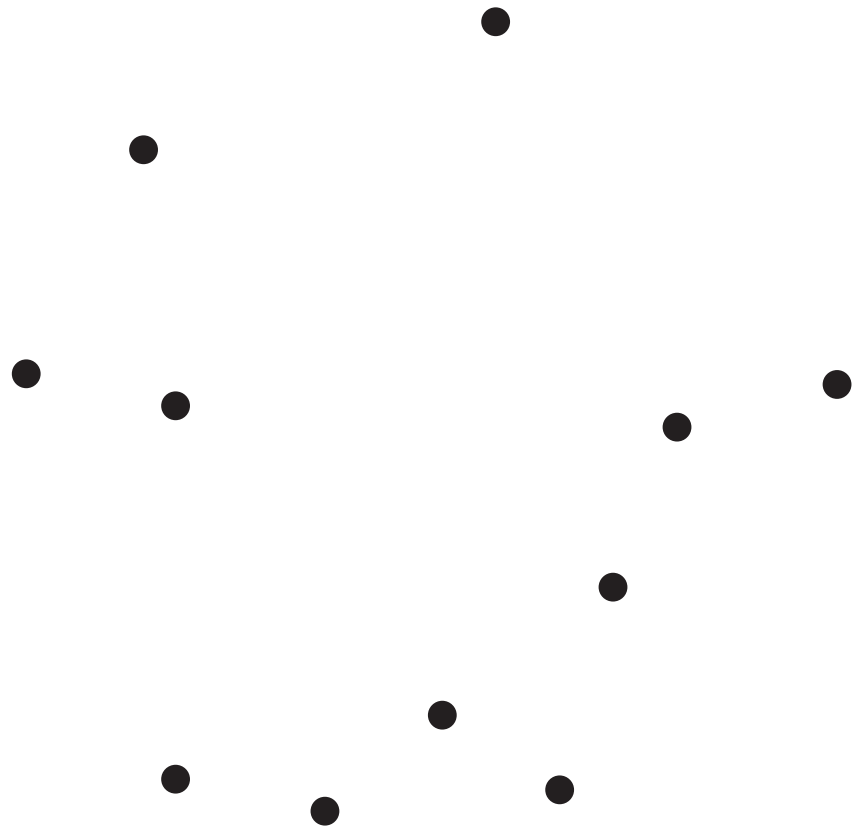
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

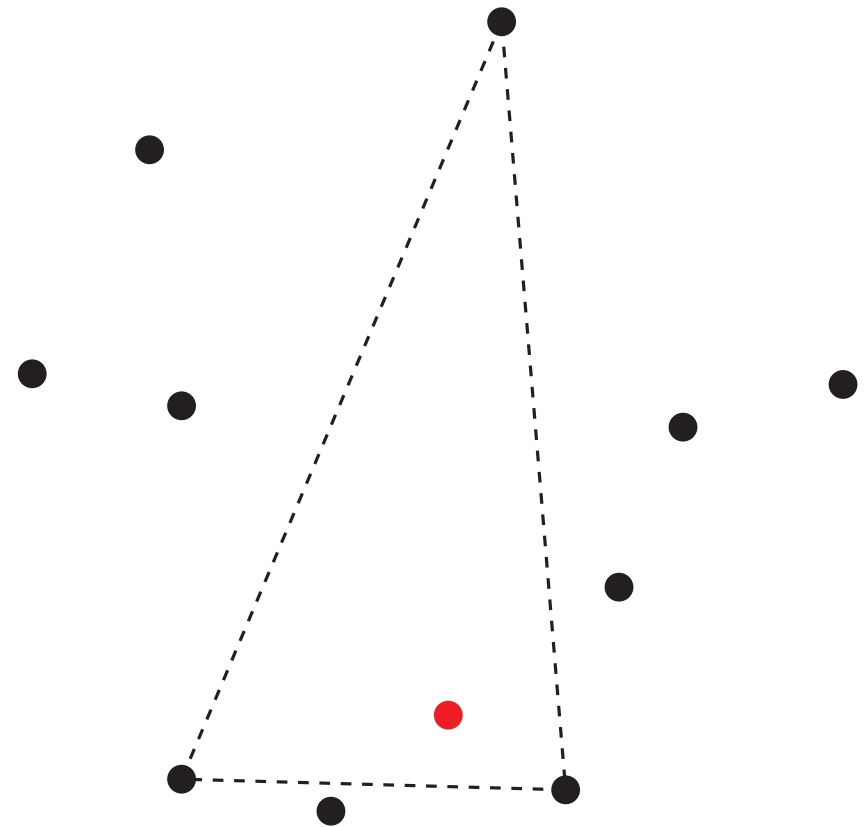
Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

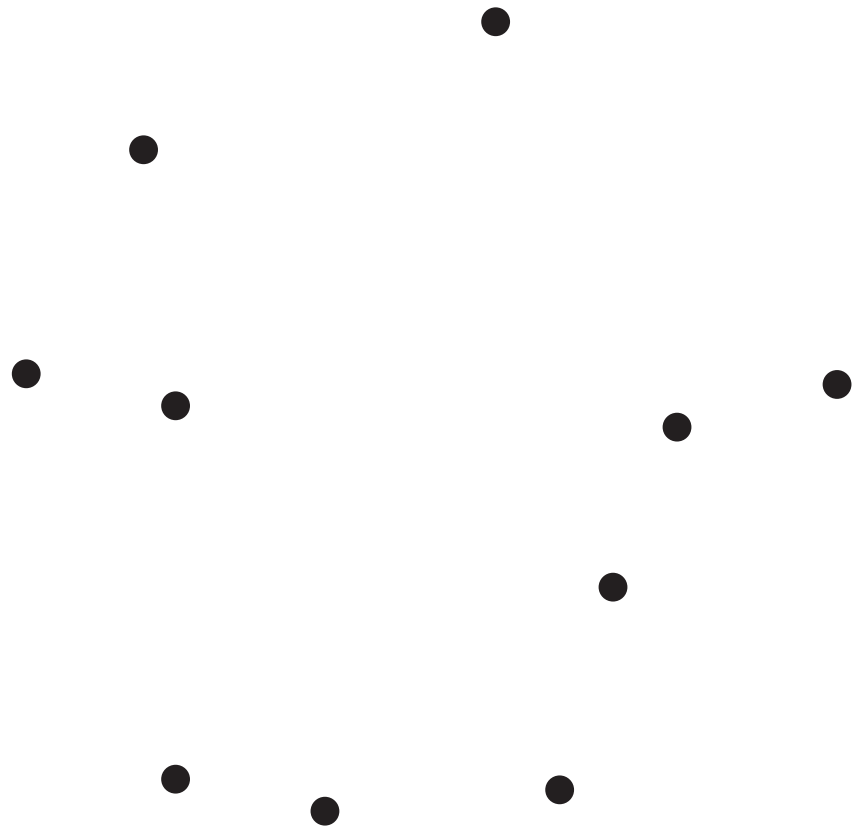
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

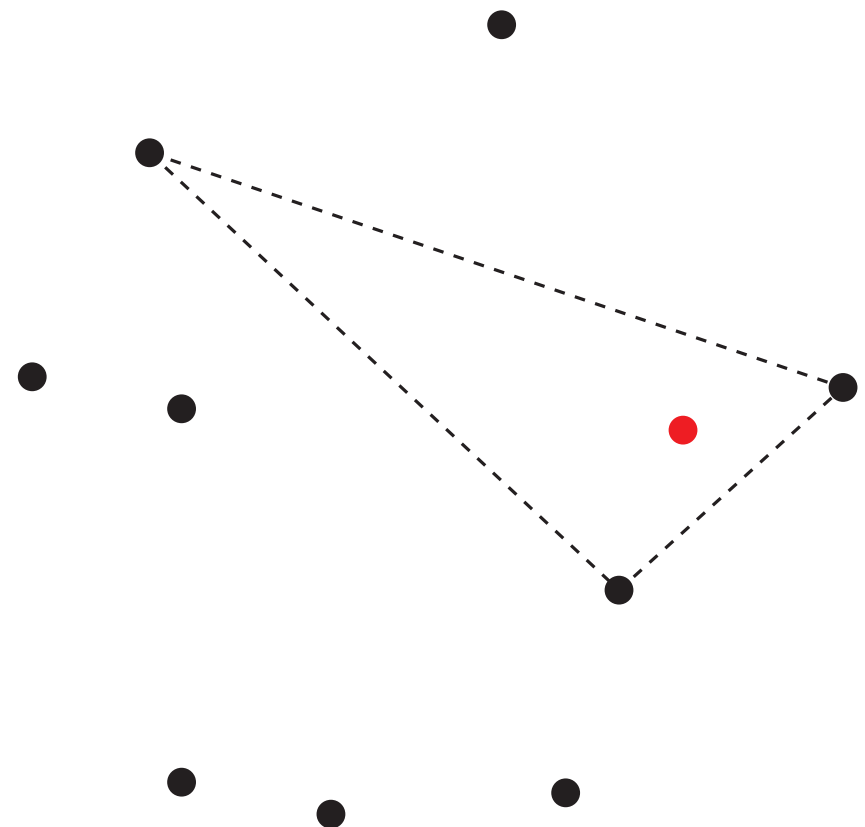
Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until all remaining points are extreme points.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Algo  
Input  
Output  
Process  
Flow  
R



# CONVEX HULL IN 2D

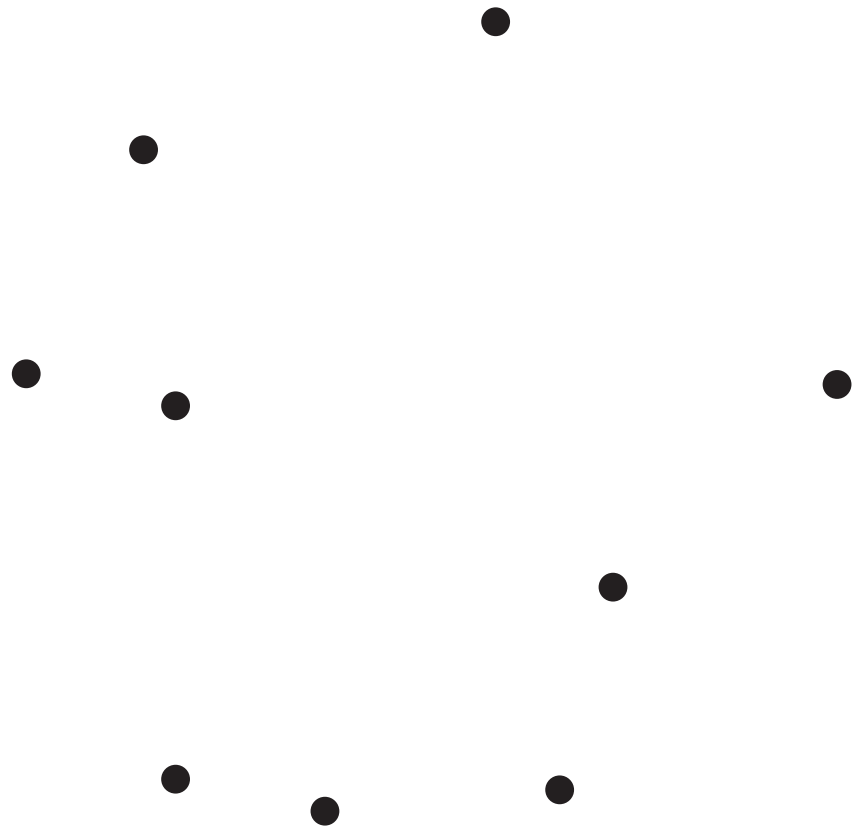
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

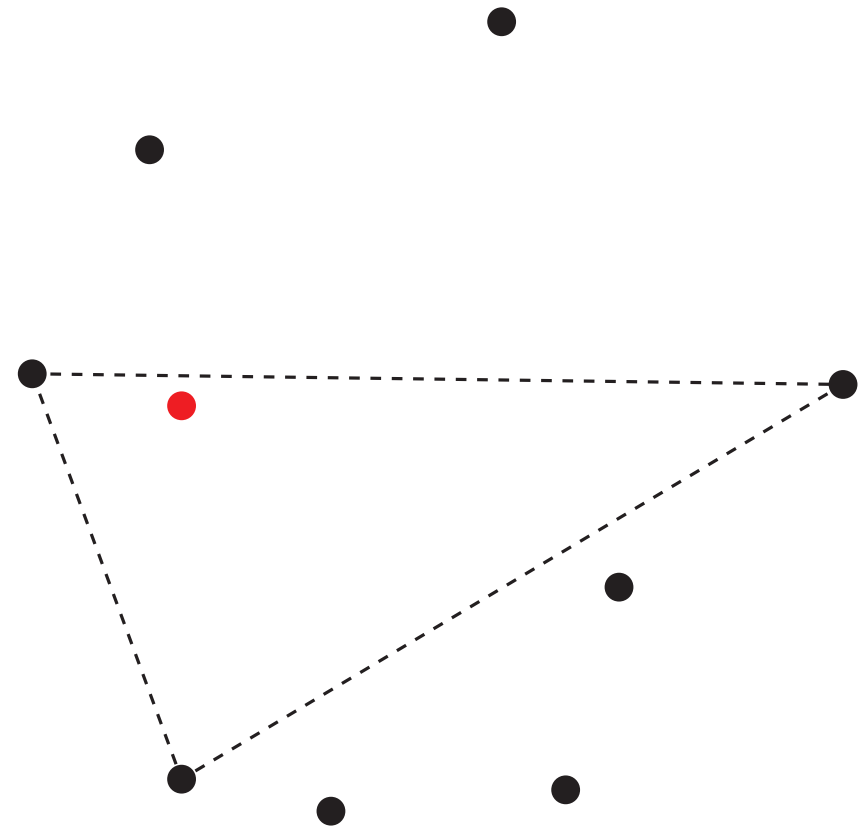
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until all remaining  $p_i$ 's are extreme points.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Process  
Flow  
R

# CONVEX HULL IN 2D

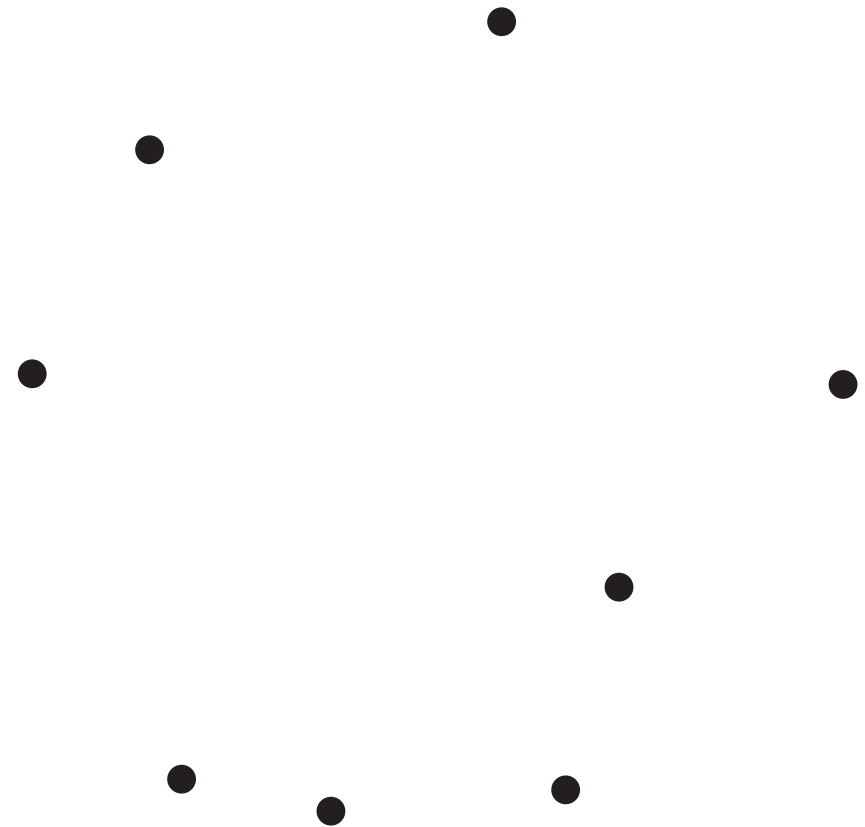
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

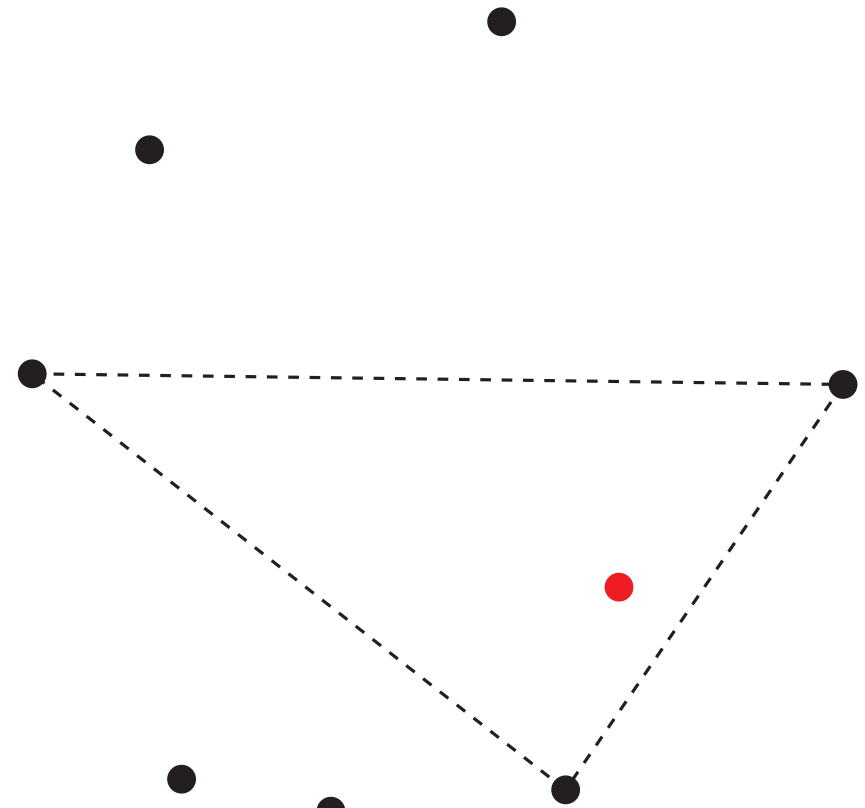
Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .



extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until all remaining points are extreme points.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Proc  
To  
R

# CONVEX HULL IN 2D

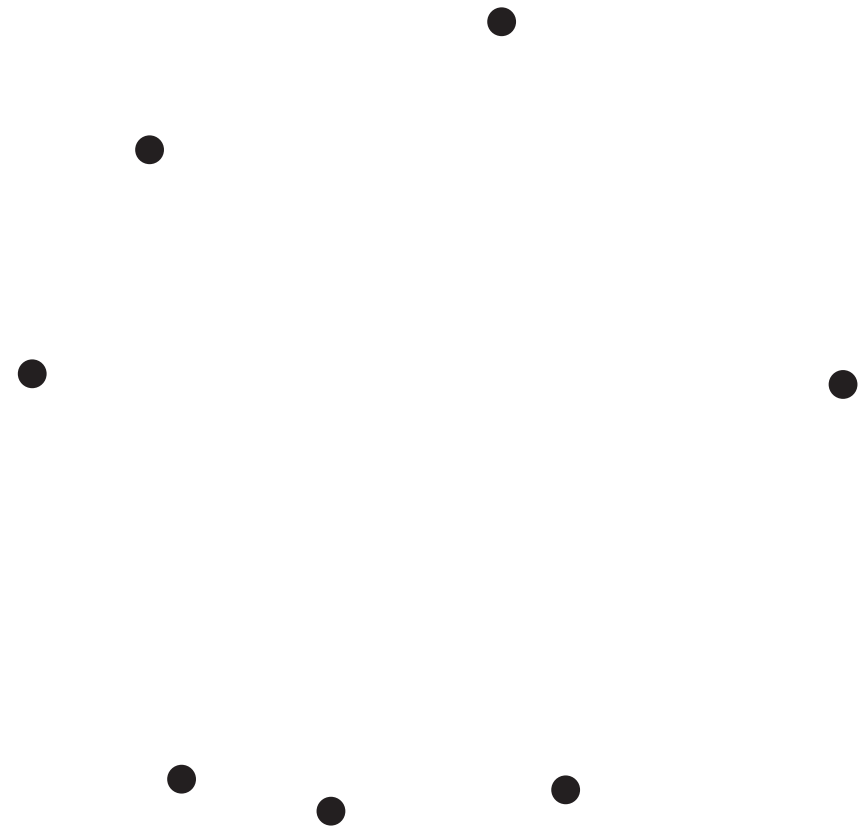
## Computing the extreme points

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Algo  
Input  
Output  
Process  
Flow  
R

# CONVEX HULL IN 2D

## Computing the extreme points

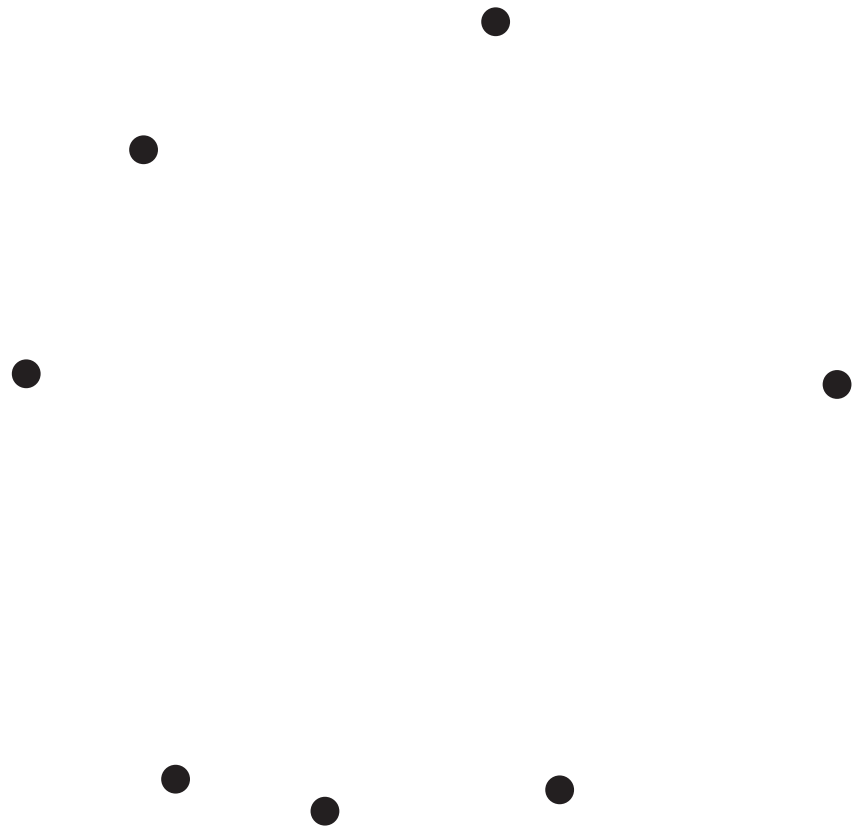
### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the point  $p_i$  belongs to the boundary of the convex hull of  $X$  if and only if  $p_i$  does not lie in any of the triangles  $p_j p_k p_l$ .

extreme points

$l \neq i$ ,  
if  $p_i$  lies in the triangle  $p_j, p_k, p_l$ , eliminate  $p_i$ .  
Repeat until no more triangles of surviving  $p_i$ 's.

$\Theta(n^4)$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
-- --  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
--  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

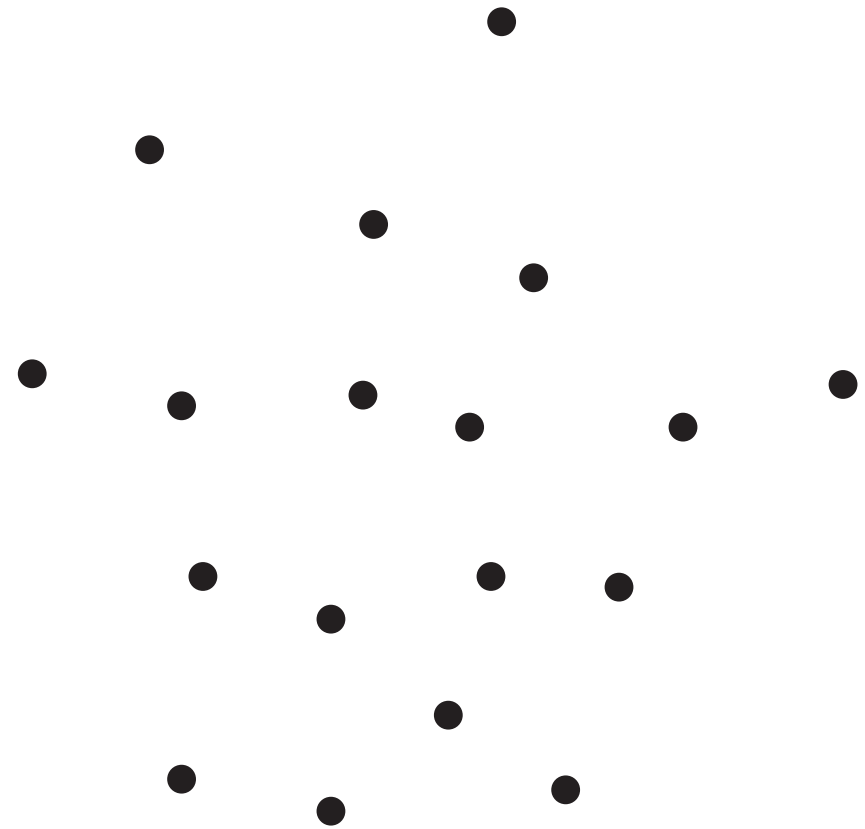


# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



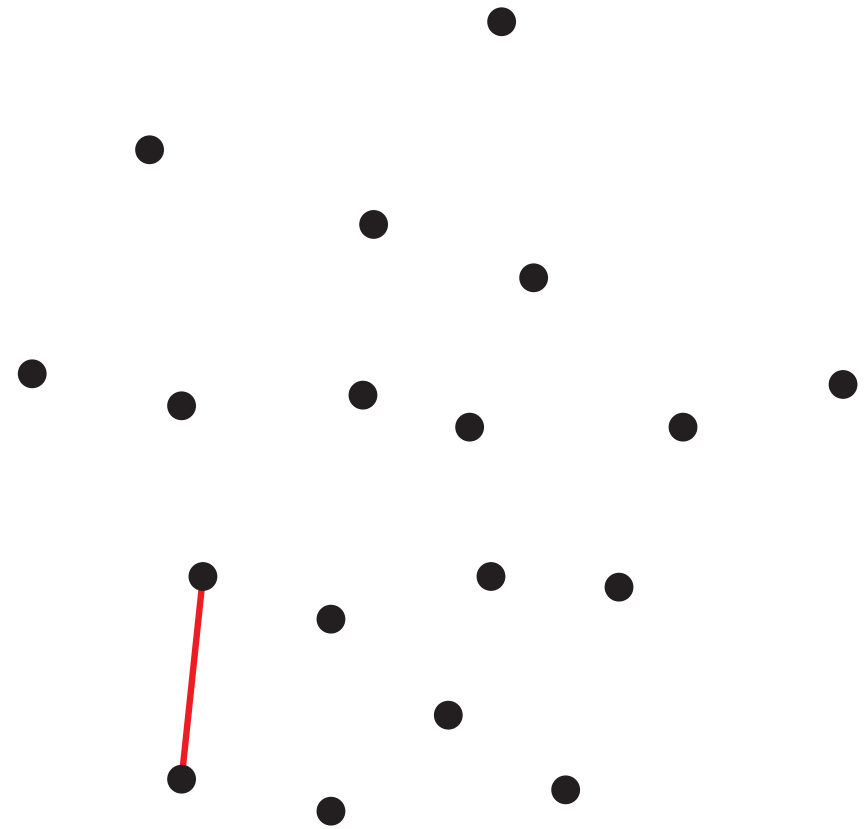
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
-- --  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



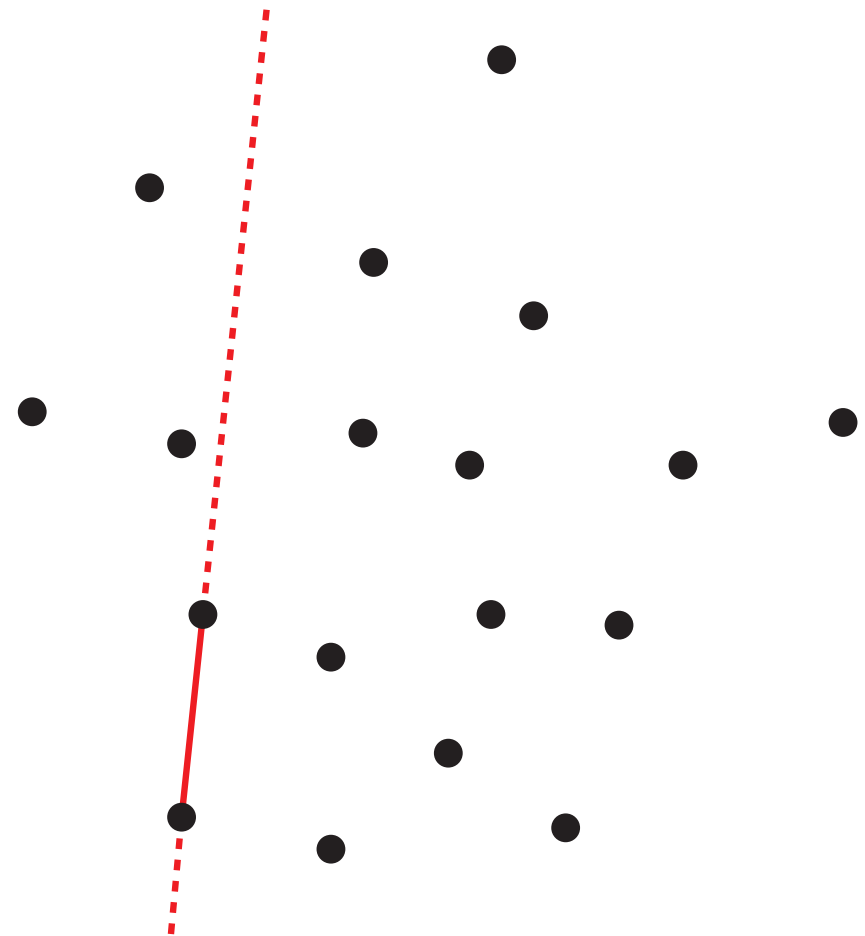
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



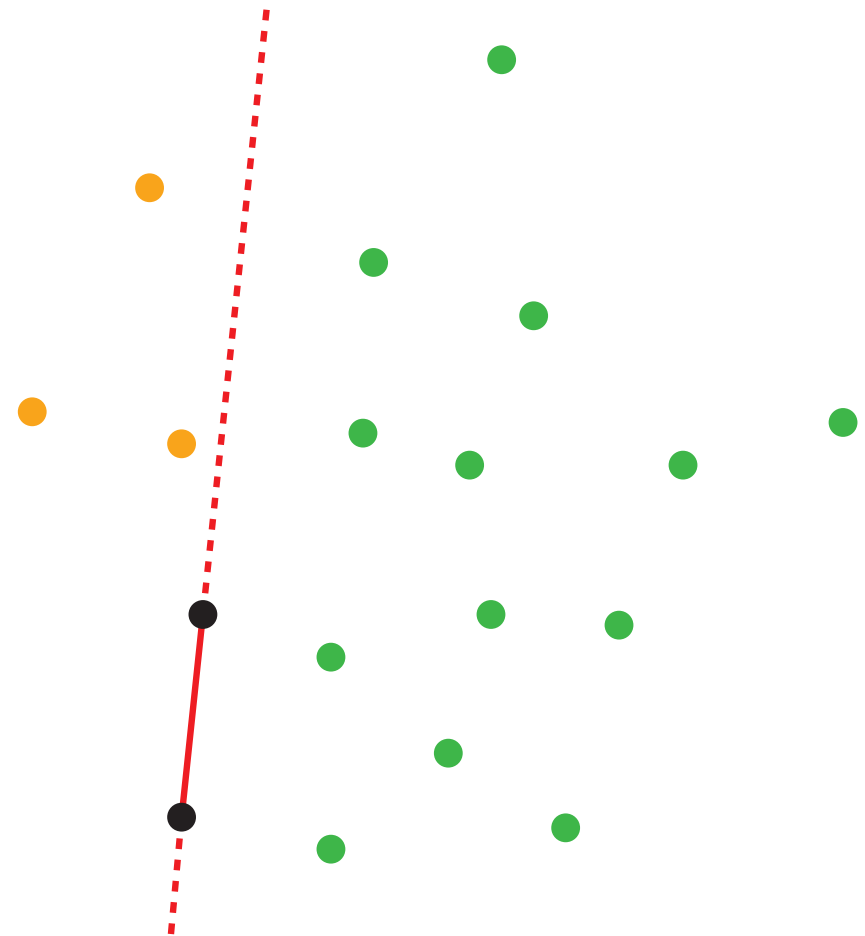
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



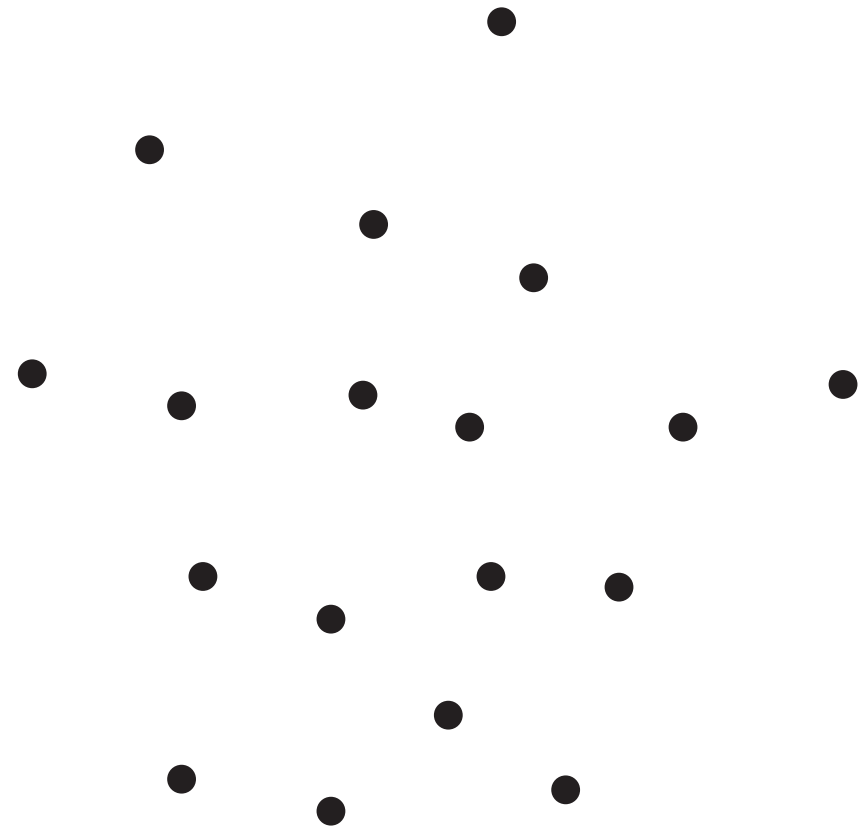
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



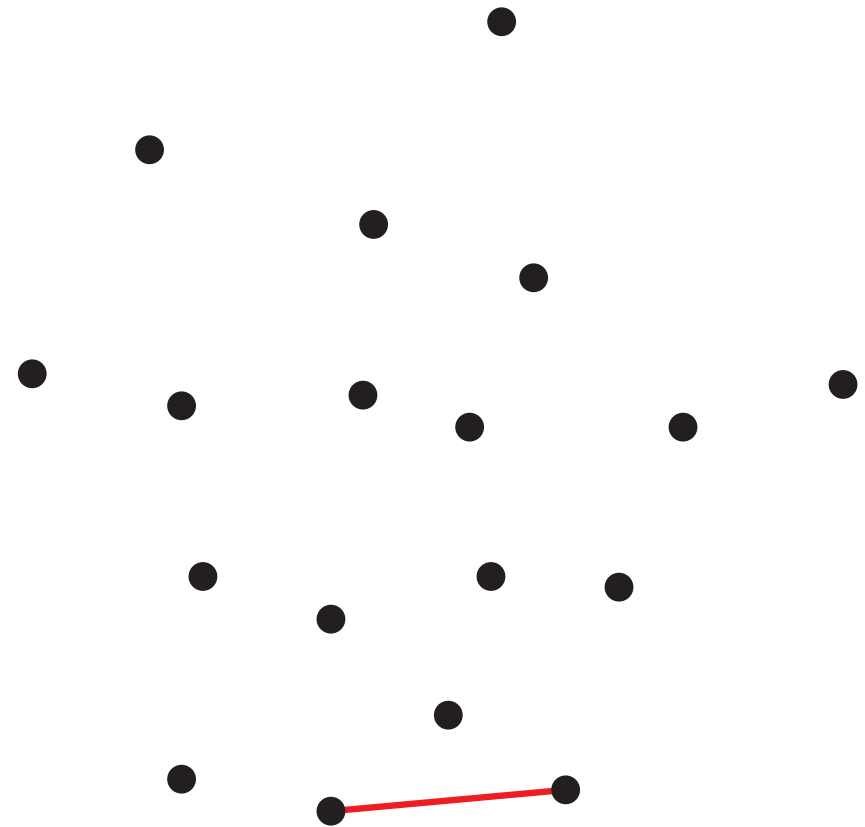
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



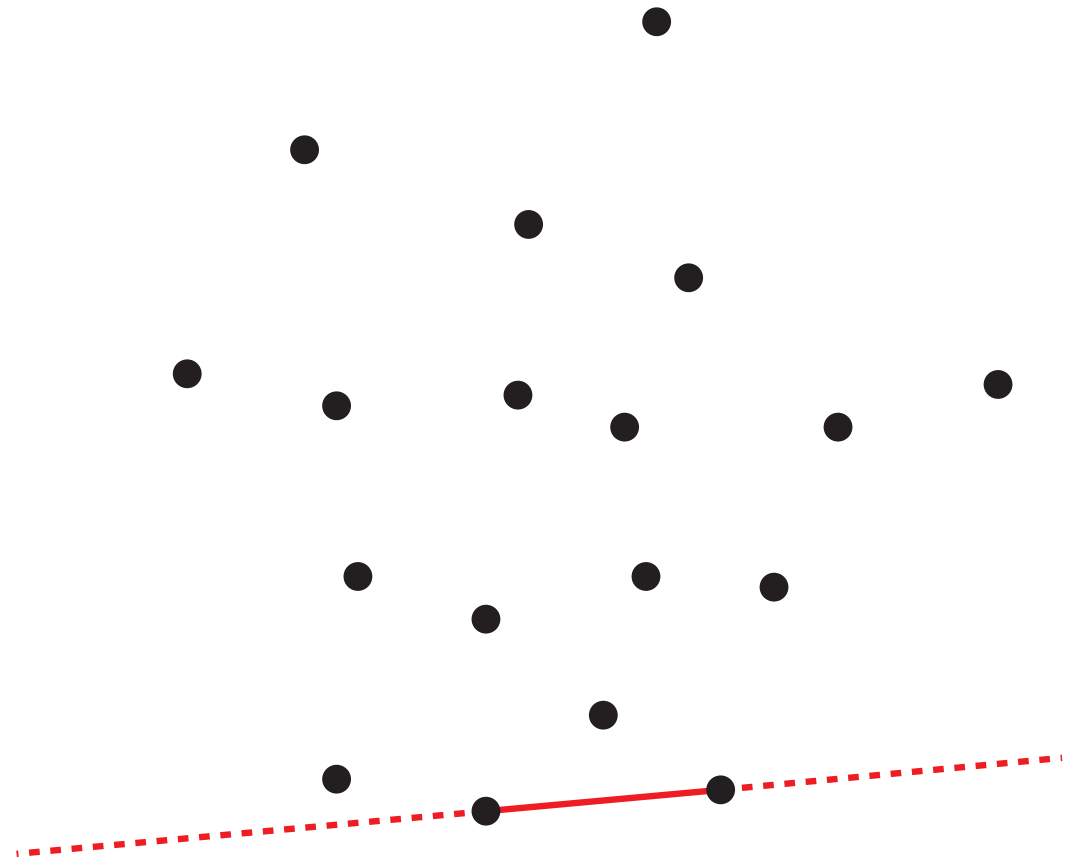
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



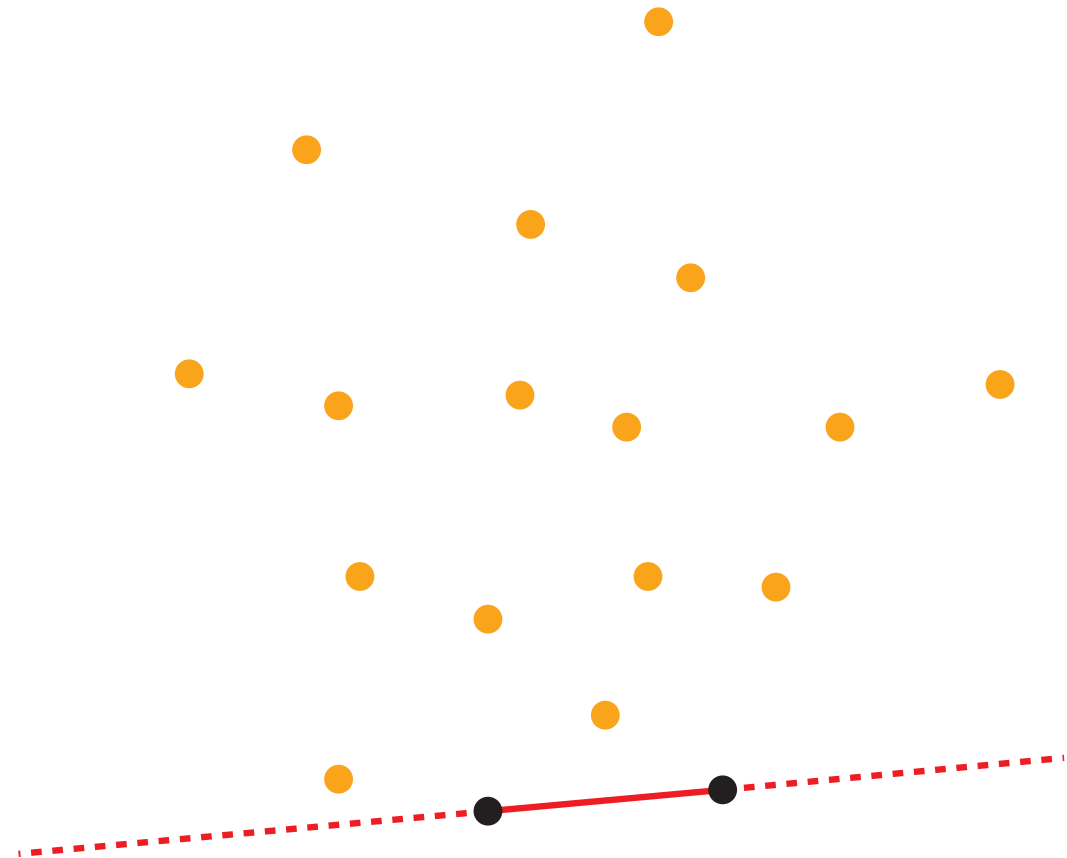
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# CONVEX HULL IN 2D

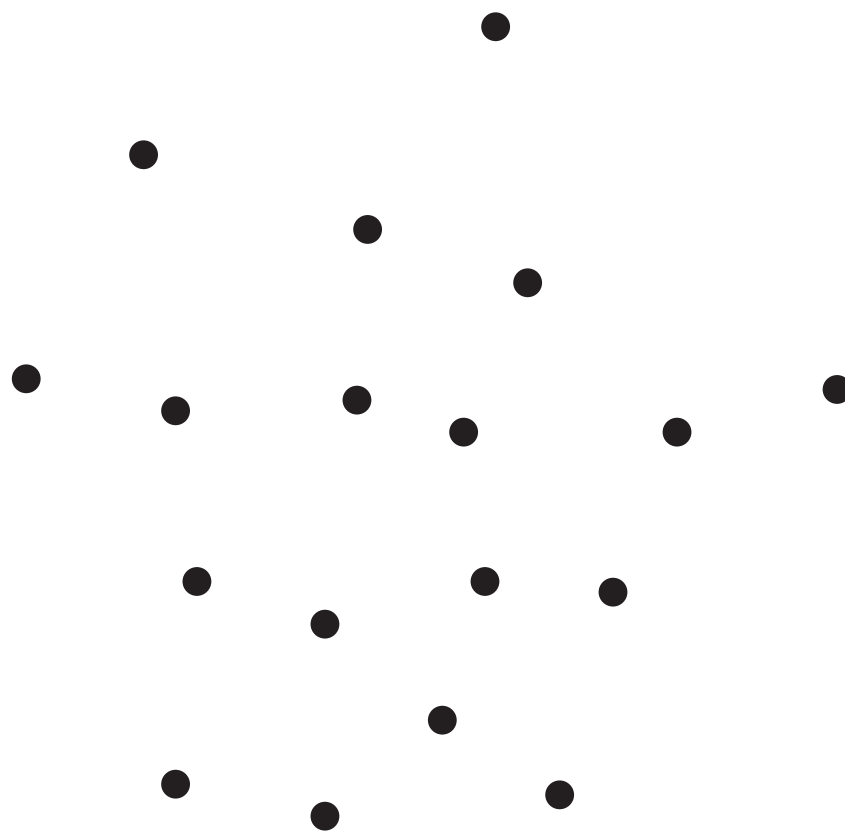
## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .

extreme segments

for all  $p_k$  with  $k \neq i, j$   
the halfplane defined by  $p_i p_j$ .  
ative, return the segment  $p_i p_j$ .



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

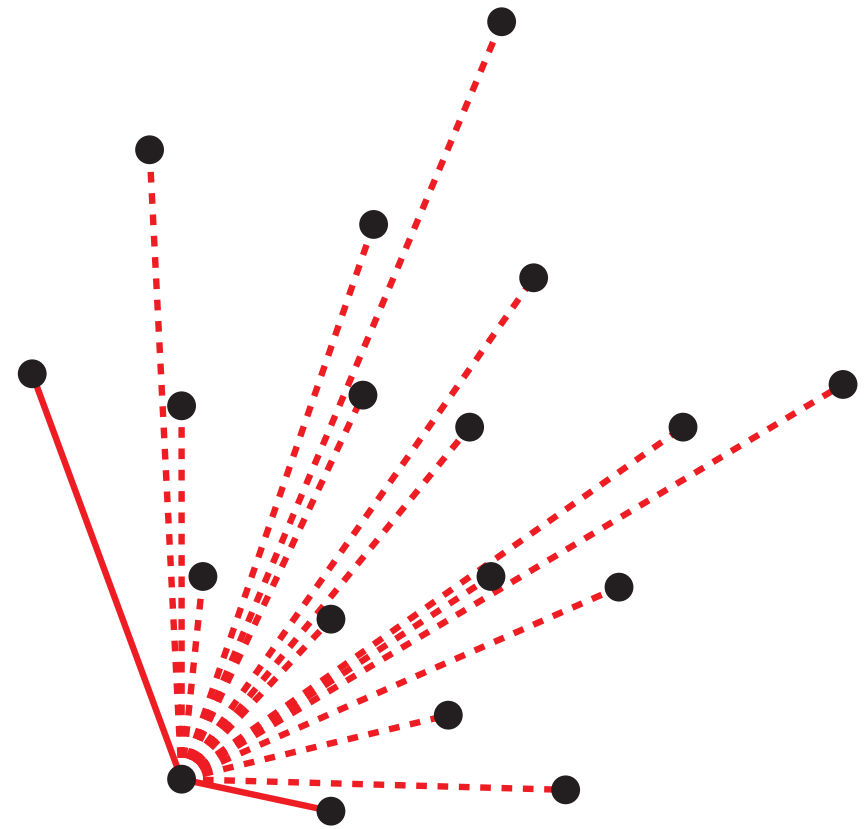
## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .

extreme segments

for all  $p_k$  with  $k \neq i, j$   
the halfplane defined by  $p_i p_j$ .  
ative, return the segment  $p_i p_j$ .



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Algo  
Input  
Output  
Proc  
To

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .

Algo

Input

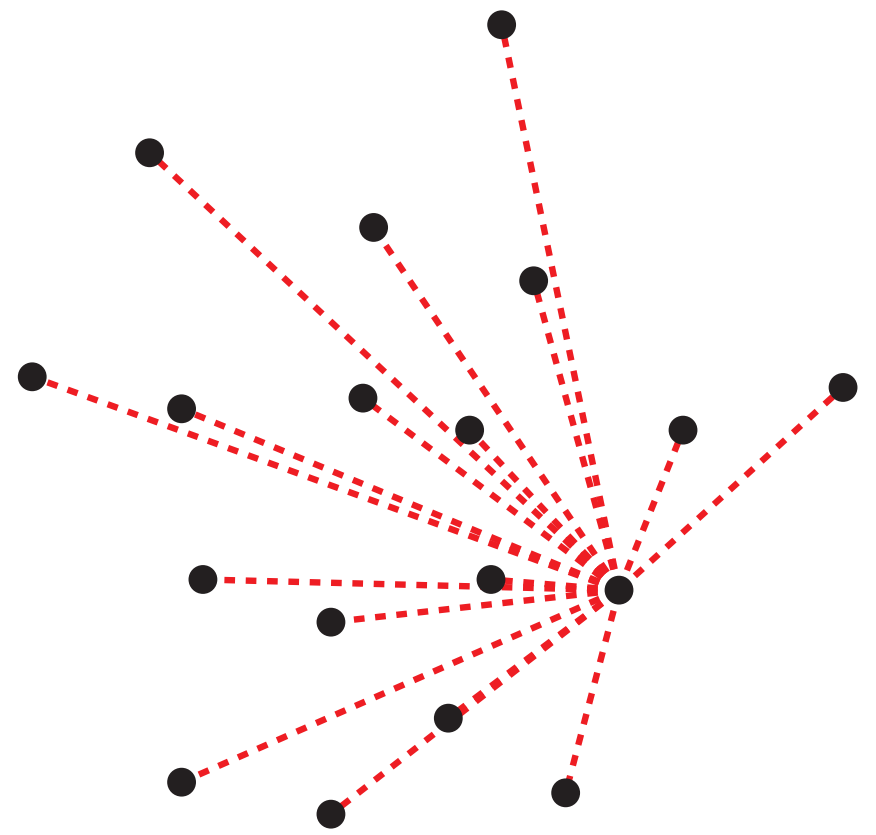
Output

Proc

Time

extreme segments

for all  $p_k$  with  $k \neq i, j$   
in the halfplane defined by  $p_i p_j$ .  
if not, return the segment  $p_i p_j$ .



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

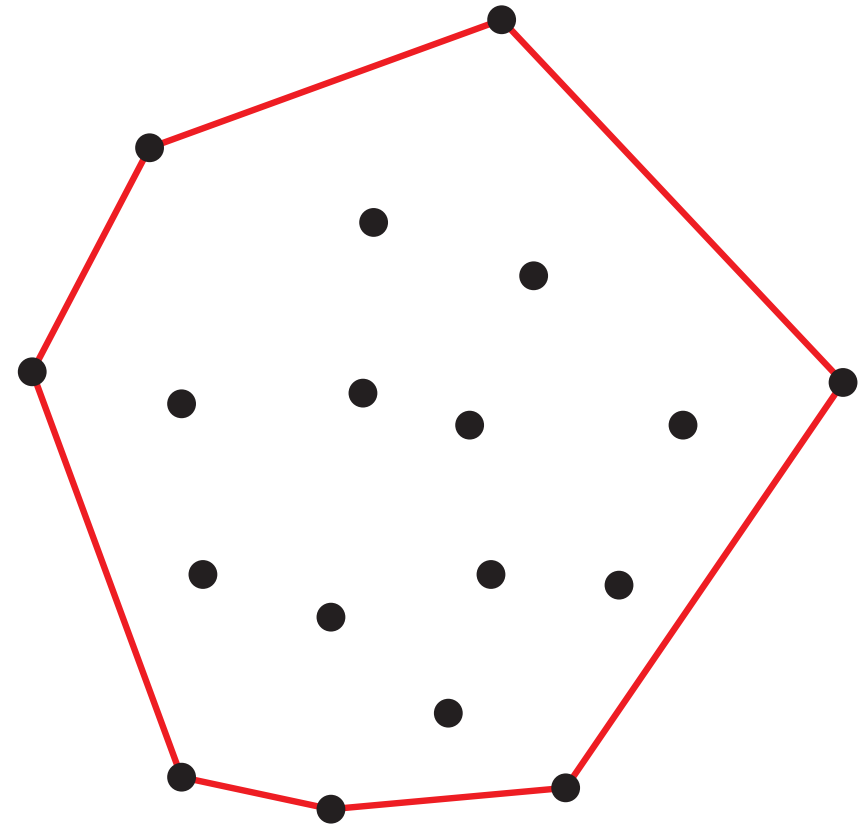
www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



extreme segments

for all  $p_k$  with  $k \neq i, j$   
the halfplane defined by  $p_i p_j$ .  
ative, return the segment  $p_i p_j$ .

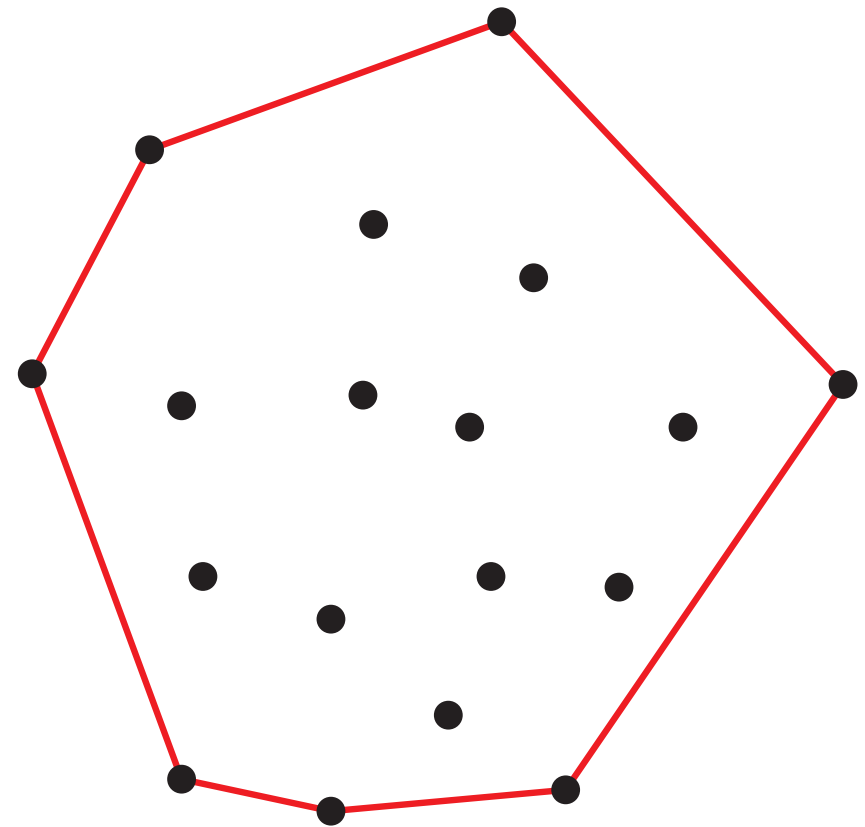
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the extreme segments

### Characterization

Given  $X = \{p_1, \dots, p_n\}$ , the segment  $p_i p_j$  is an extreme segment if and only if all the points  $p_k$  with  $k \neq i, j$  lie in the same halfplane defined by the line  $p_i p_j$ .



extreme segments

for all  $p_k$  with  $k \neq i, j$   
the halfplane defined by  $p_i p_j$ .  
ative, return the segment  $p_i p_j$ .

$\Theta(n^3)$

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Computing the convex hull

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

Computing the convex hull (sorted list of its vertices)

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
-- --  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Computing the convex hull

**Input:**

$P = \{p_1, \dots, p_n\} \subset \mathbb{R}^2$  a set of  $n$  points in the plane

of the vertices of  $ch(P)$  sorted in counterclockwise order

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
--  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# CONVEX HULL IN 2D

## Computing the convex hull

**Input:**

$P = \{p_1, \dots, p_n\} \subset \mathbb{R}^2$  a set of  $n$  points in the plane

of the vertices of  $ch(P)$  sorted in counterclockwise order

$\dots, p_n\}$ , the segment  $p_i p_j$  is an edge of the convex hull of  $X$  if and only if all  $k \neq i, j$  lie to the left of the oriented line  $p_i p_j$ .

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70  
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

Carvis march

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

--

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

2. If  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

3. Add  $p_j$  to  $l$

2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

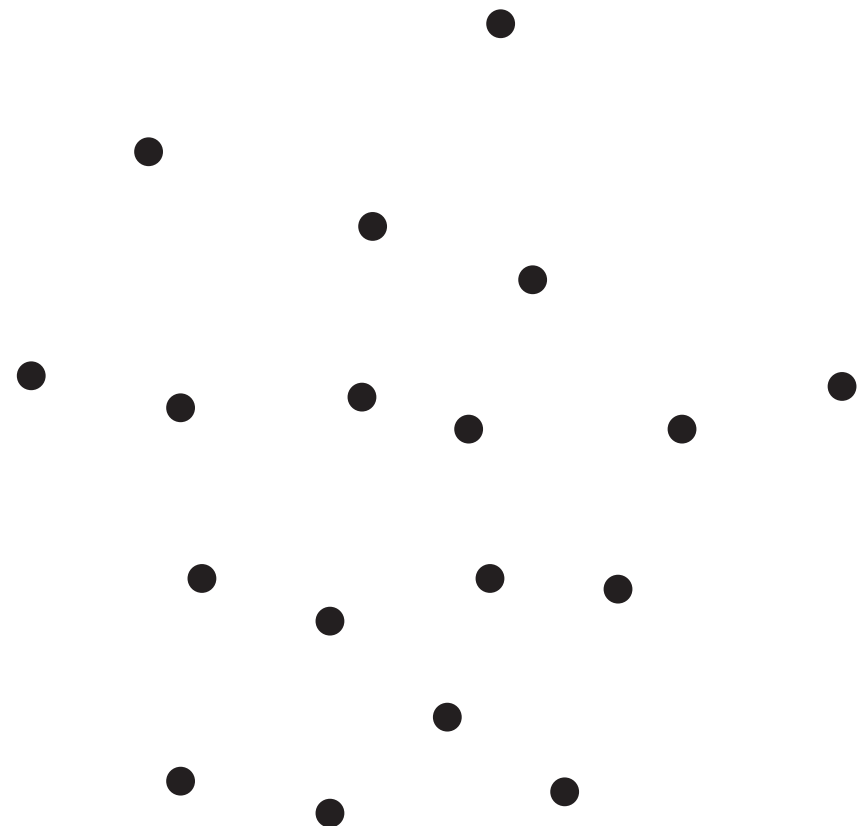
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

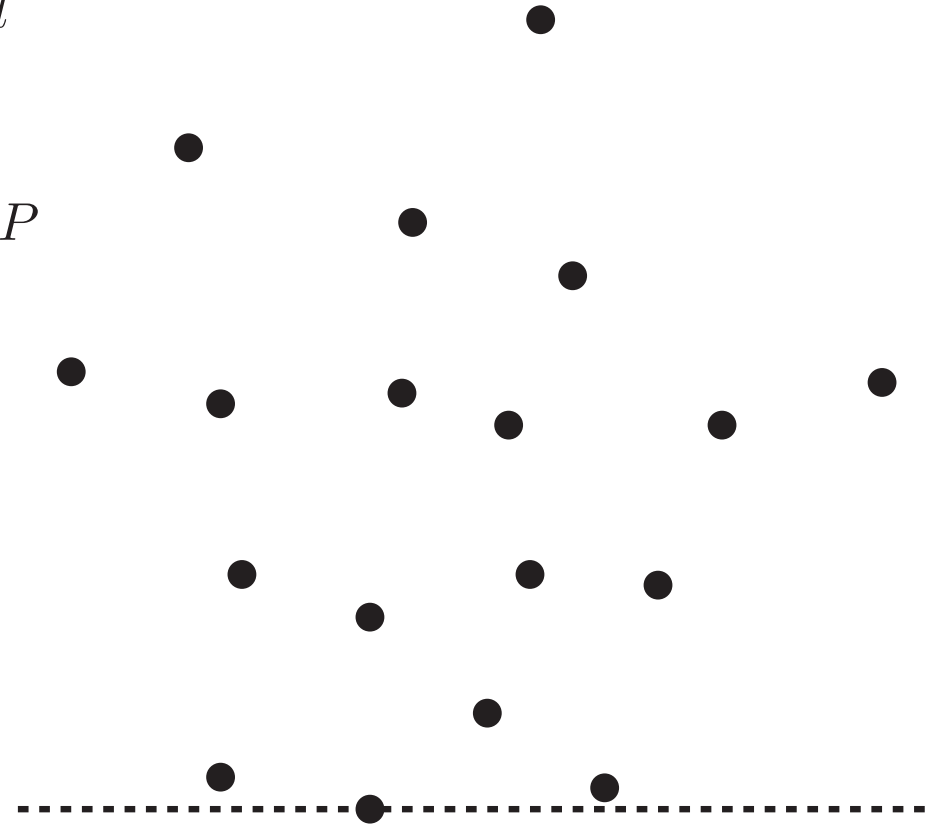
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

2. If  $Last(l) \neq First(l)$ , do:

2.1. Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

2.2. Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

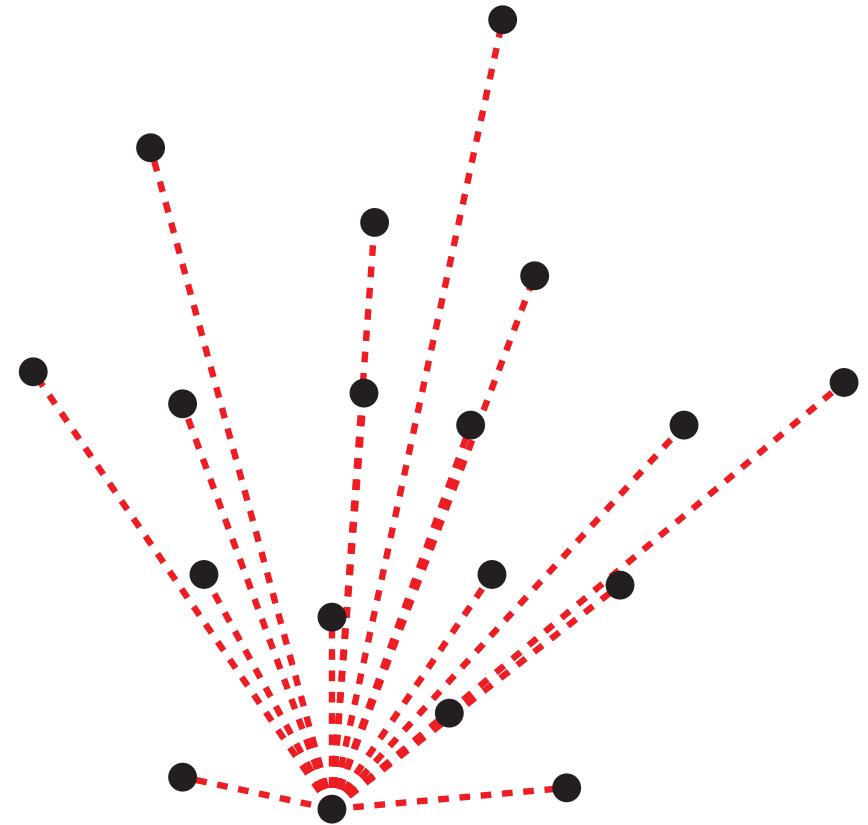
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

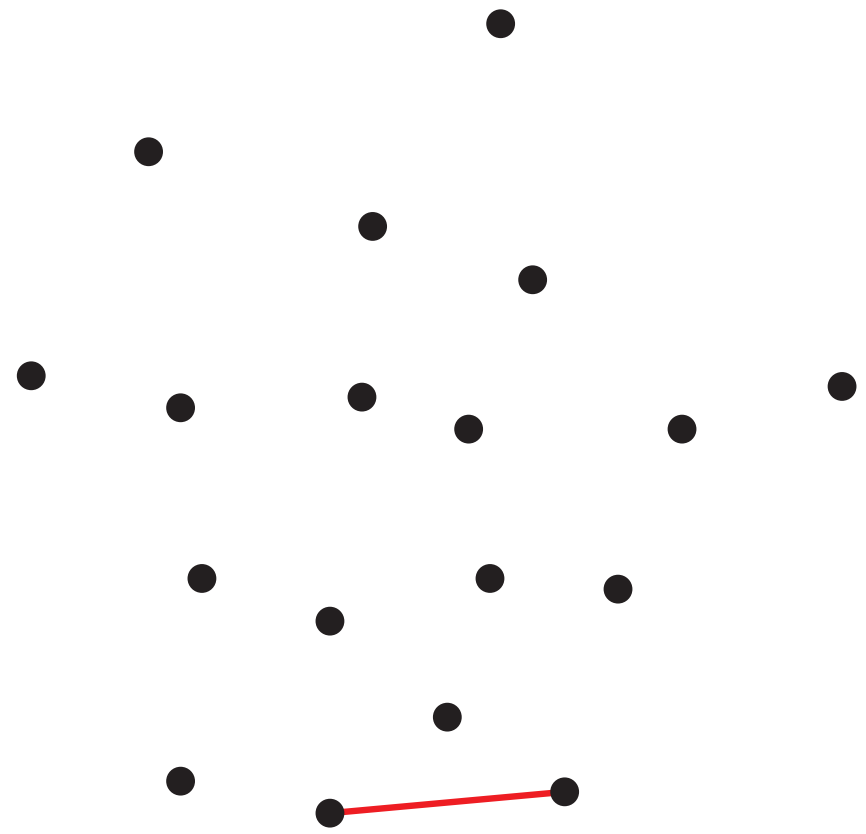
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

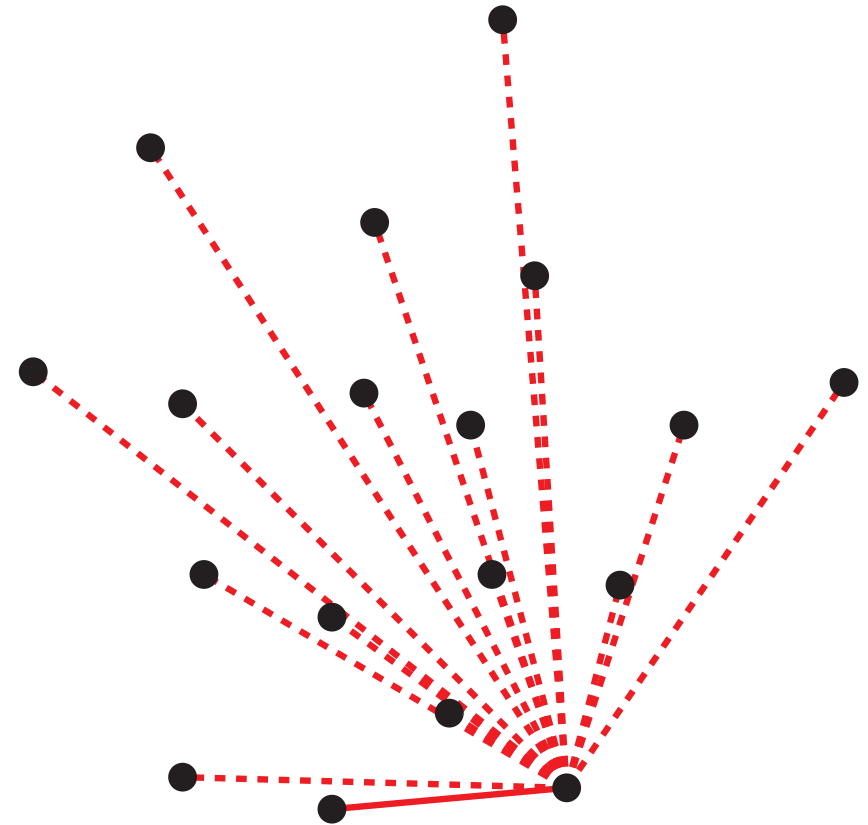
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

---



# CONVEX HULL IN 2D

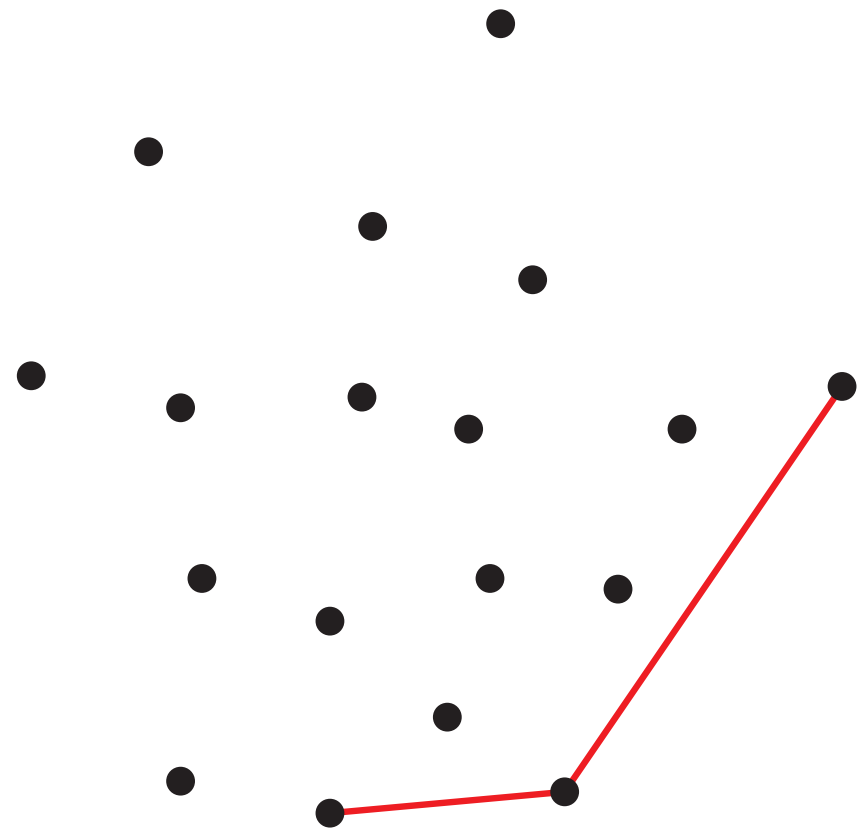
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

---

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

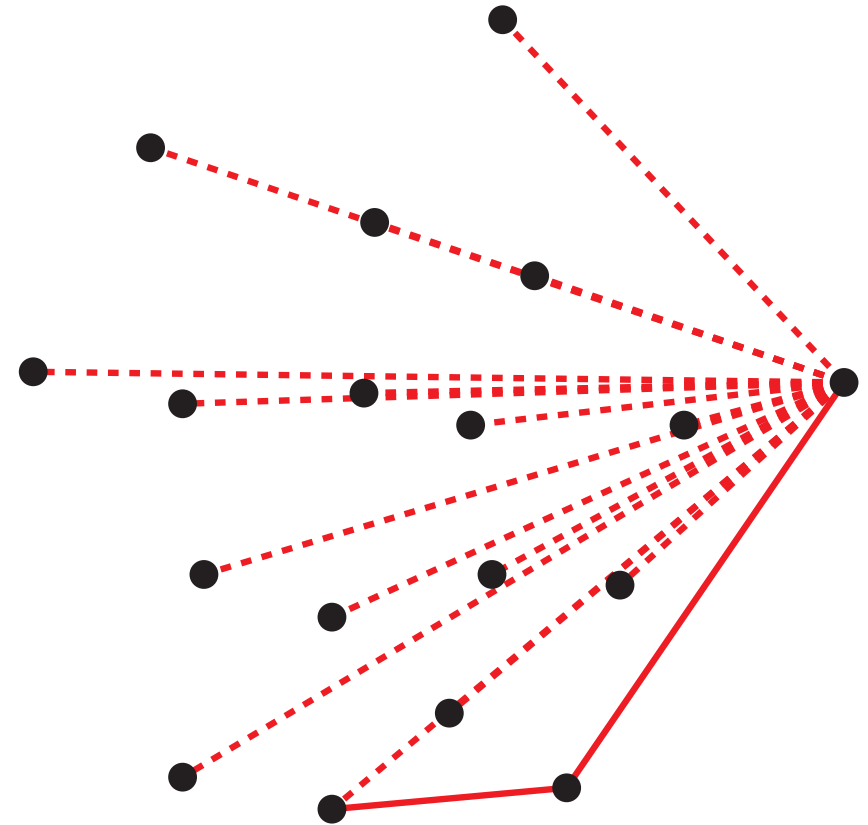
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

2. If  $Last(l) \neq First(l)$ , do:

2.1. Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

2.2. Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

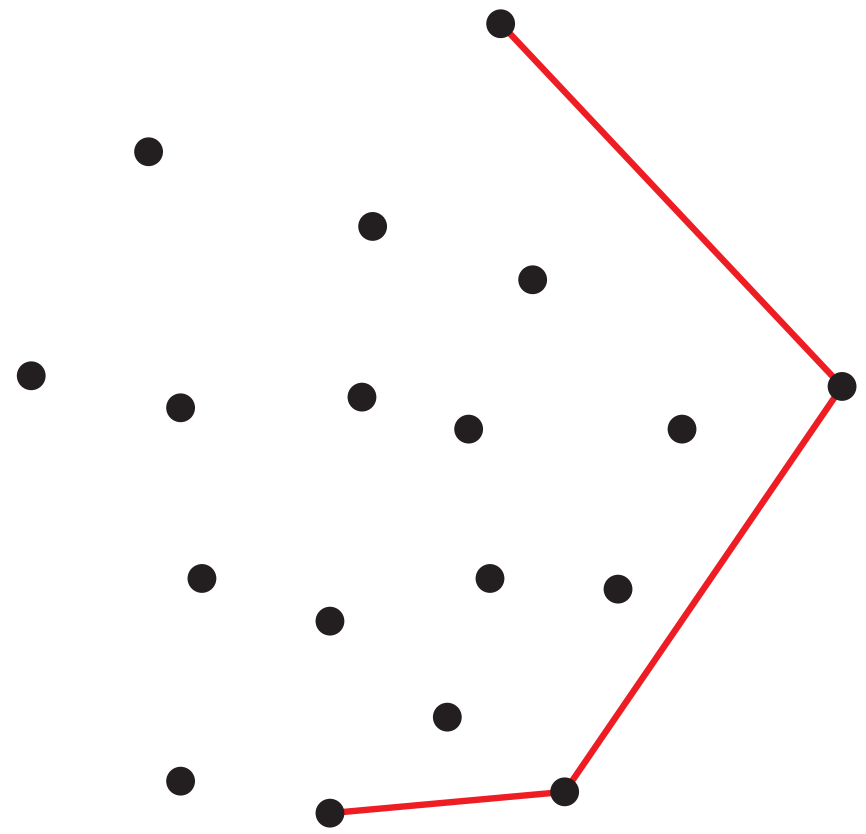
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

---

# CONVEX HULL IN 2D

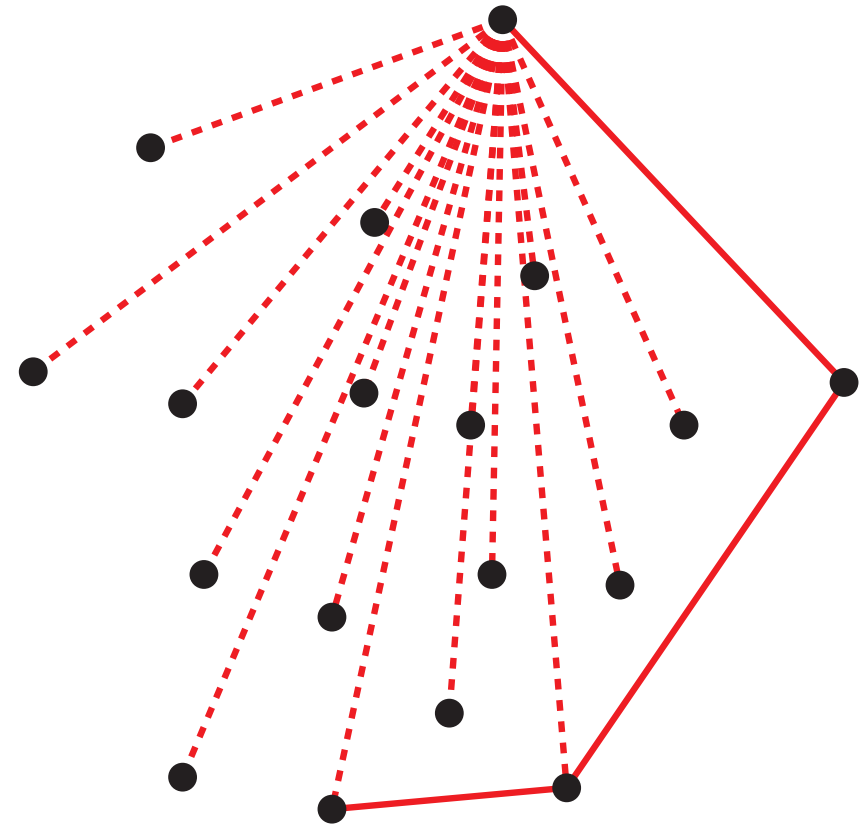
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

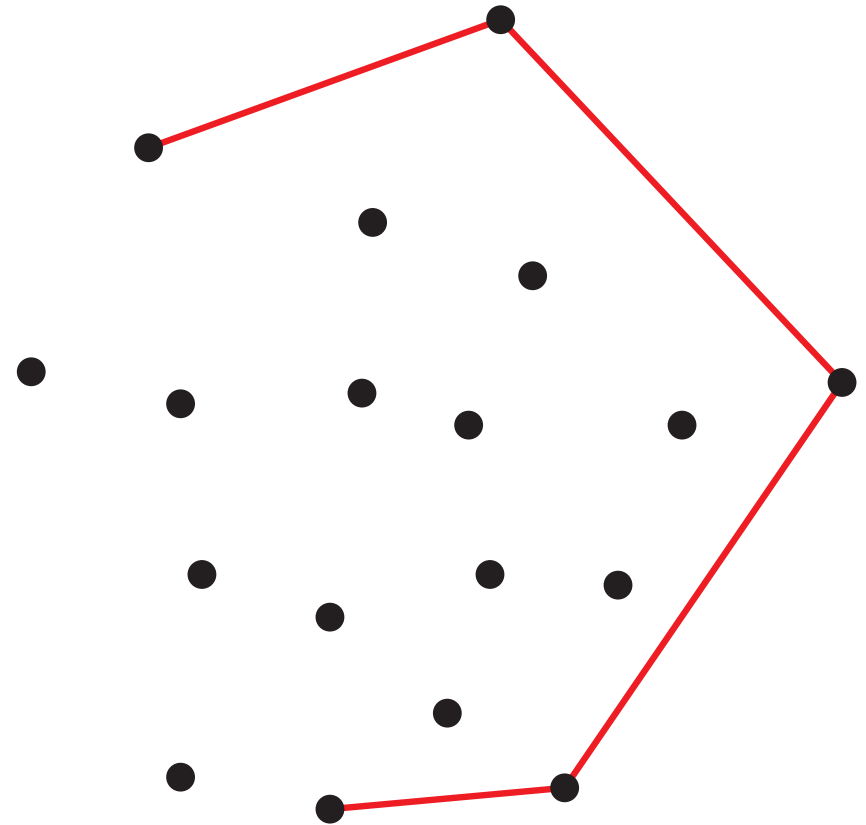
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

---

# CONVEX HULL IN 2D

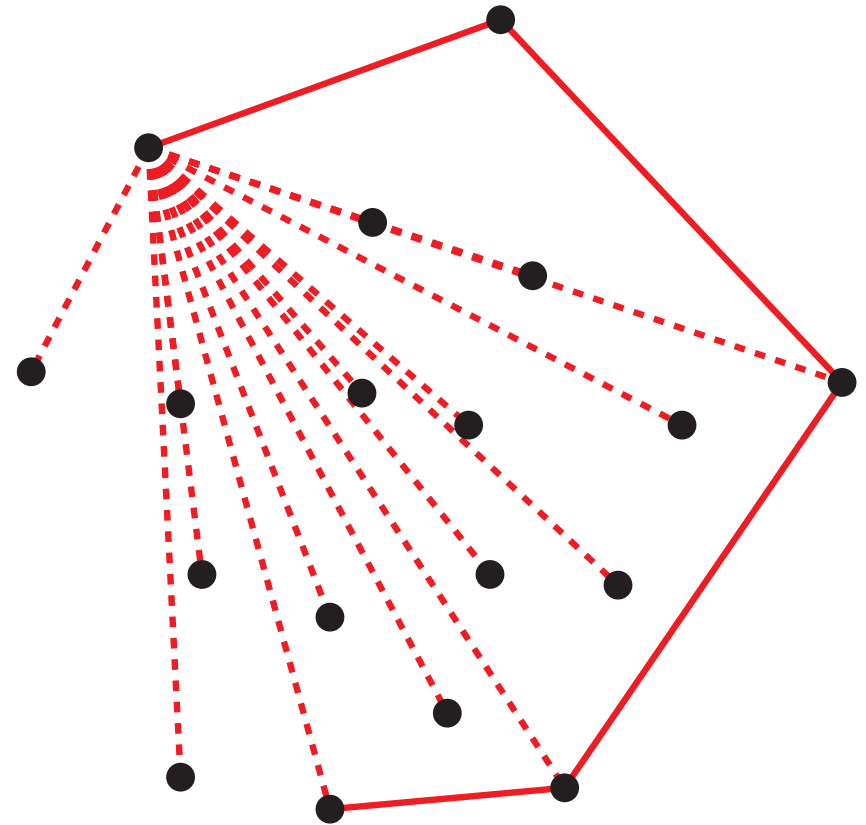
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

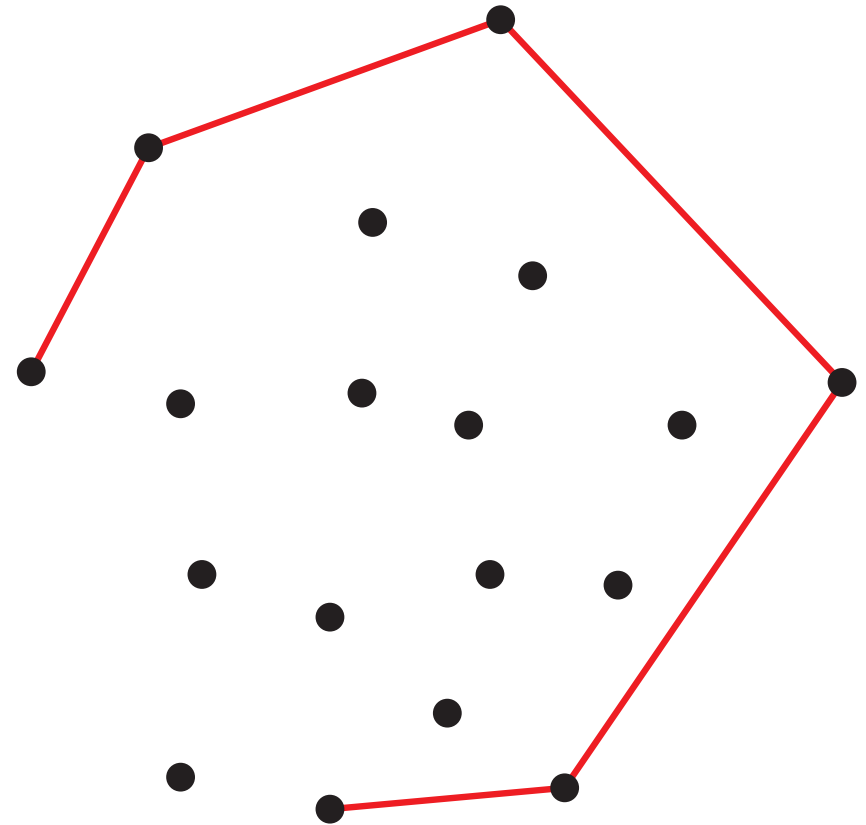
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

---

# CONVEX HULL IN 2D

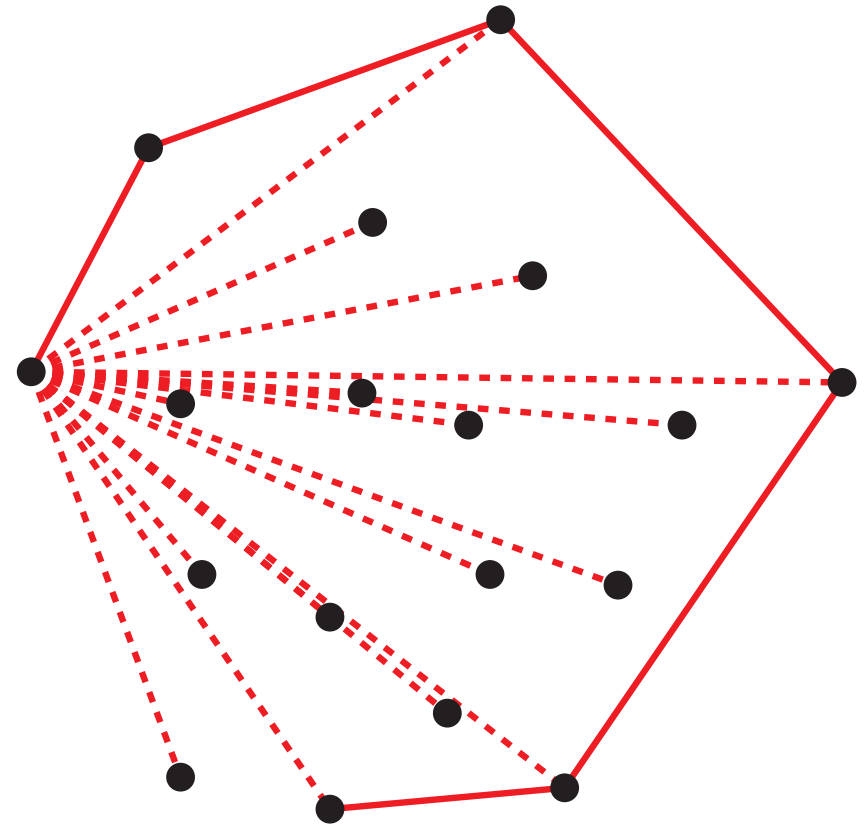
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

2. If  $Last(l) \neq First(l)$ , do:

    Find the angularly rightmost point  $p_j \in P$   
    with respect to  $v$ .

    Add  $p_j$  to  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

2

3



# CONVEX HULL IN 2D

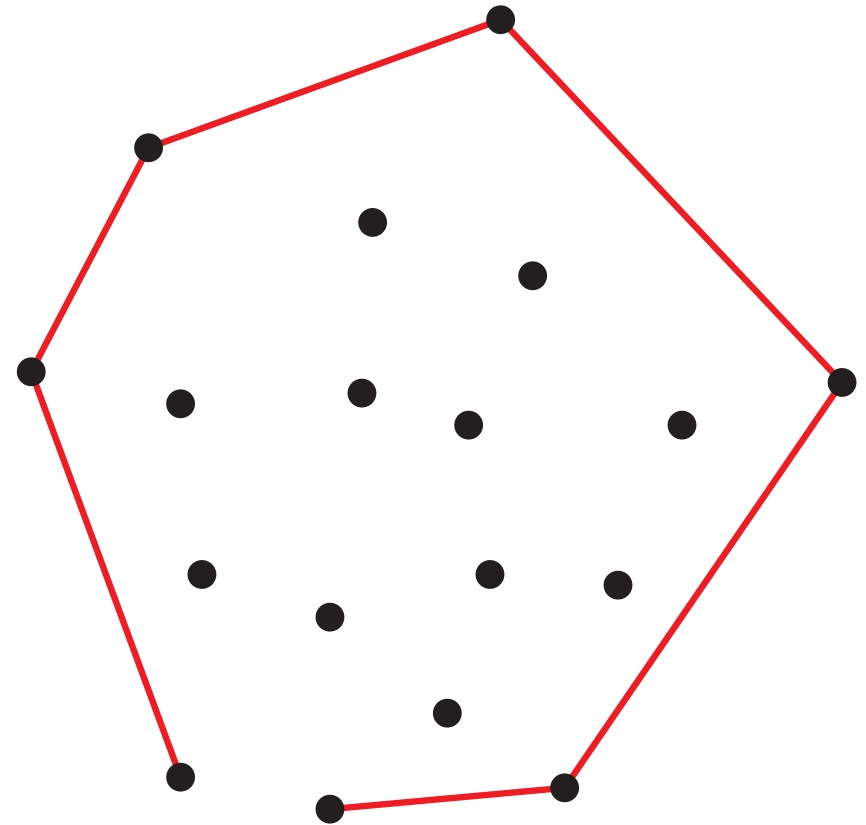
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

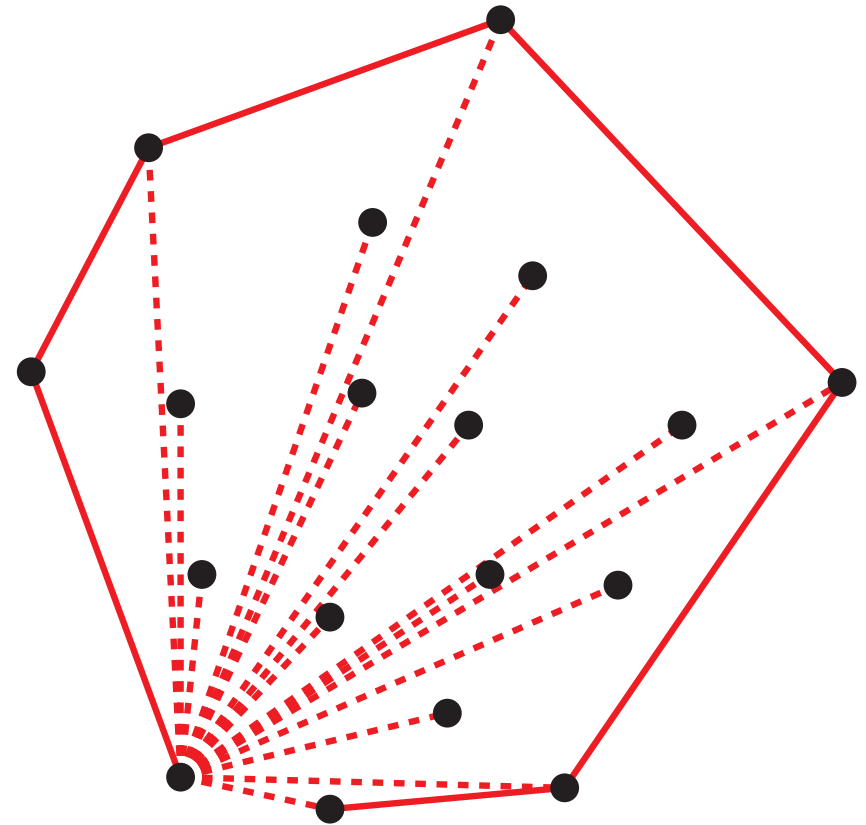
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

2. If  $Last(l) \neq First(l)$ , do:

    Find the angularly rightmost point  $p_j \in P$   
    with respect to  $v$ .

    Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

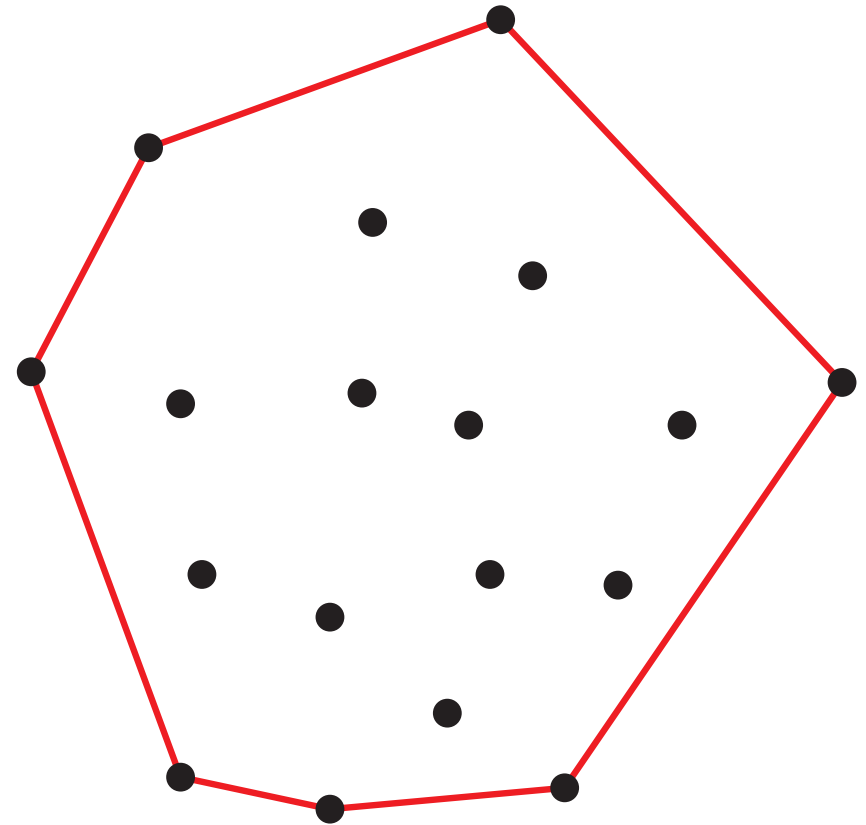
## Jarvis march

1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$



2

3

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

---

# CONVEX HULL IN 2D

## Jarvis march

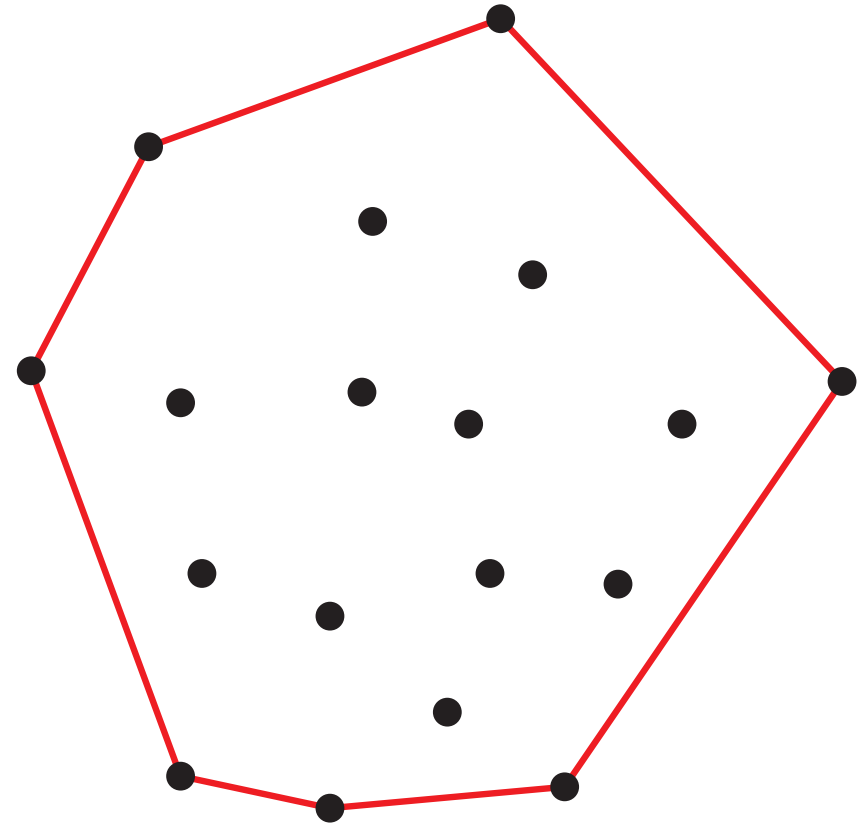
1. Find a vertex of  $ch(P)$  (for example, the lexicographically smaller point  $p_i \in P$ ) and add it to  $l$

While  $Last(l) \neq First(l)$ , do:

Find the angularly rightmost point  $p_j \in P$  with respect to  $v$ .

Add  $p_j$  to  $l$

Time complexity:  $O(n^2)$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

--

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

### Initialization

1. Find the extreme points in the horizontal and vertical directions.
2. Compute the convex hull of these (between 2 and 3) points.
3. For the remaining points, and classify them according to their position (NE, SE, SW, NW) and remove them if they lie in the interior.

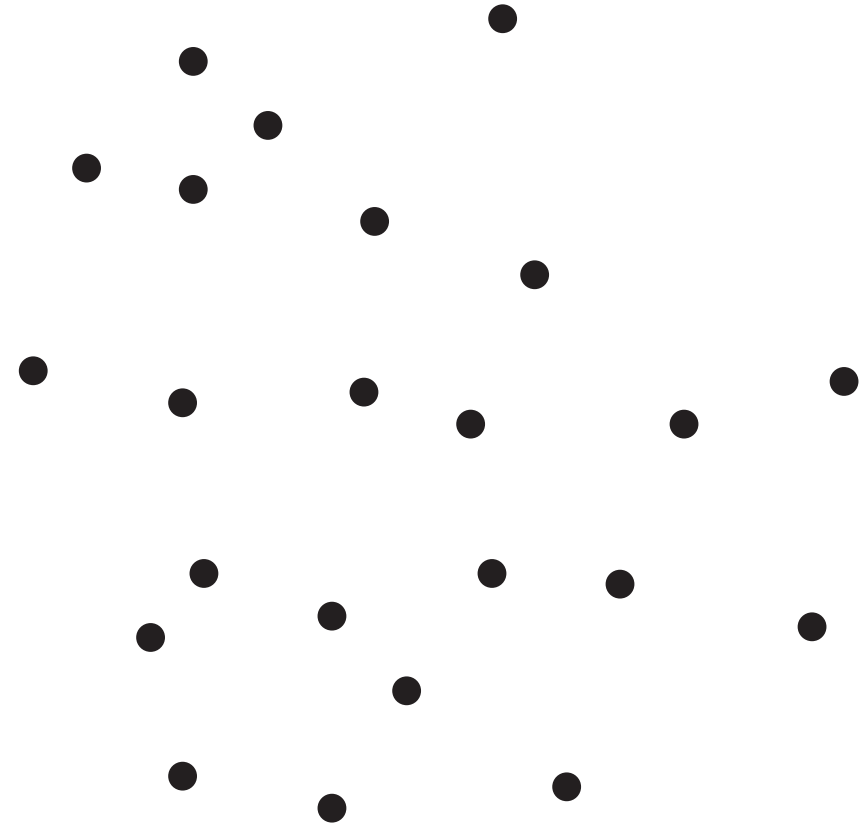
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

### Initialization

1. Find the extreme points in the horizontal and vertical directions.
2. Find the convex hull of these (between 2 and 3) points.
3. For the remaining points, and classify them according to their position (NE, SE, SW, NW) relative to the hull and remove them if they lie in the interior.



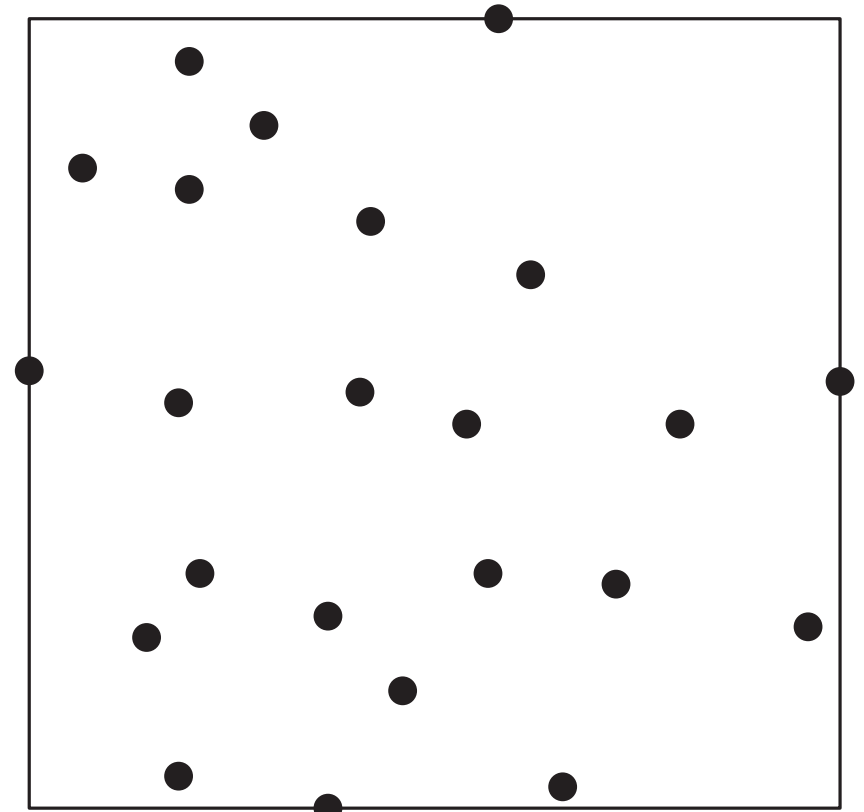
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

### Initialization

1. Find the extreme points in the horizontal and vertical directions.
2. Find the convex hull of these (between 2 points).
3. For the remaining points, and classify them according to their position (NE, SE, SW, NW) relative to the hull and remove them if they lie in the interior.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

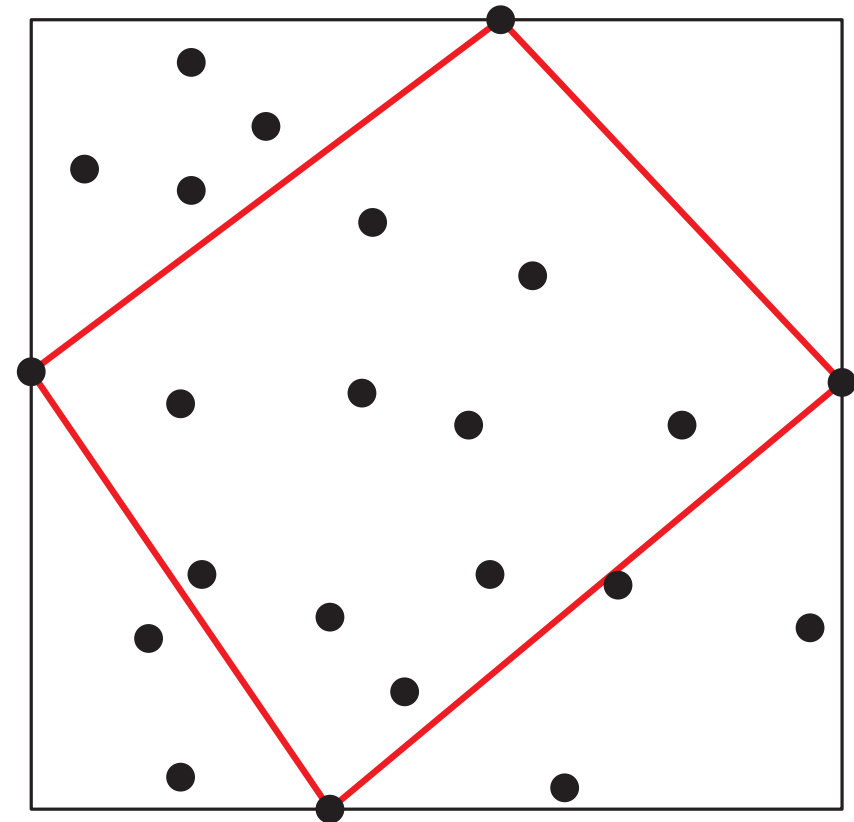


# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

### Initialization

1. Find the extreme points in the horizontal and vertical directions.
2. Find the convex hull of these (between 2 and 4) points.
3. For the remaining points, and classify them according to their position (NE, SE, SW, NW) relative to the hull and remove them if they lie in the interior.



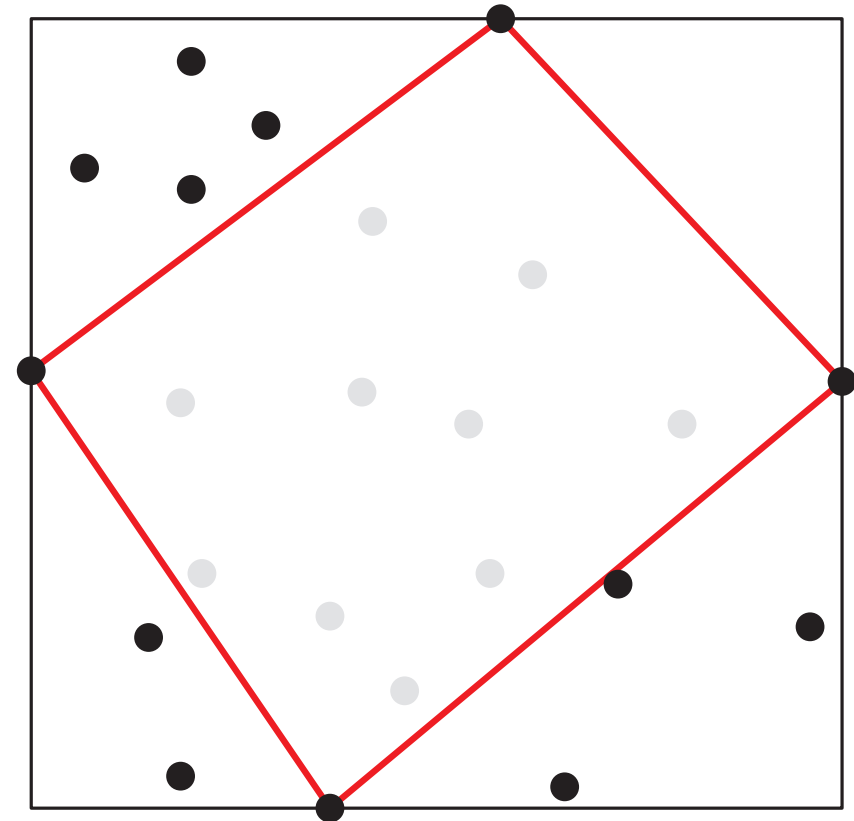
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

### Initialization

1. Find the extreme points in the horizontal and vertical directions.
2. Find the convex hull of these (between 2 and 4) points.
3. For the remaining points, and classify them according to their position (NE, SE, SW, NW) relative to the hull and remove them if they lie in the interior.



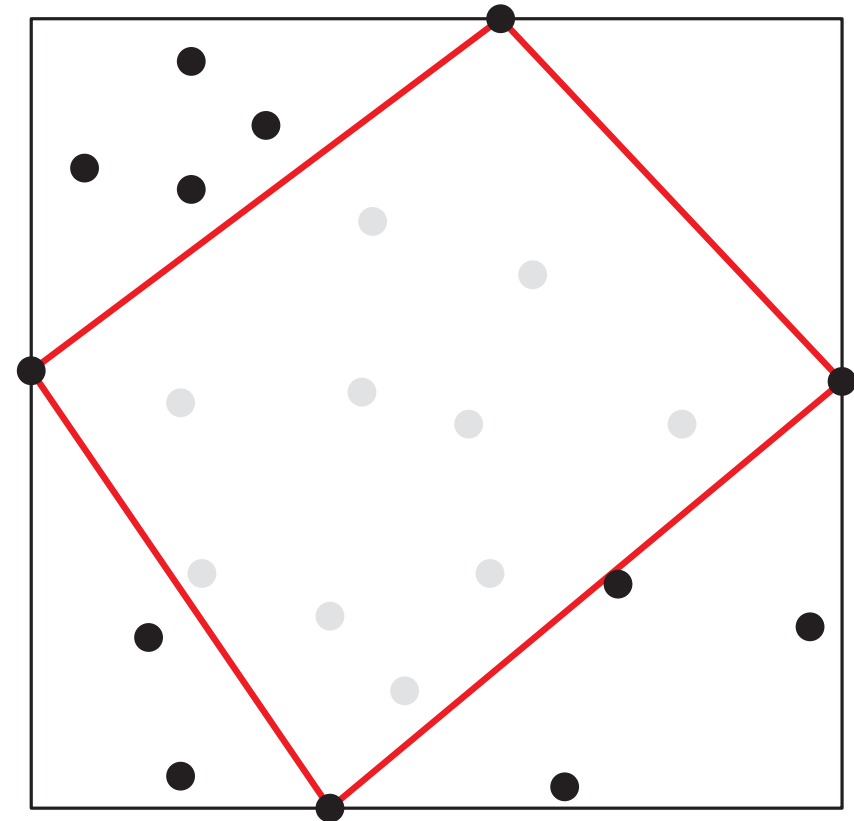
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

### Initialization

1. Find the extreme points in the horizontal and vertical directions.
2. Find the convex hull of these (between 2 and 4) points.
3. For the remaining points, and classify them according to their position (NE, SE, SW, NW) relative to the hull edges. Prune them if they lie in the interior.



Run this step:  $O(n)$

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

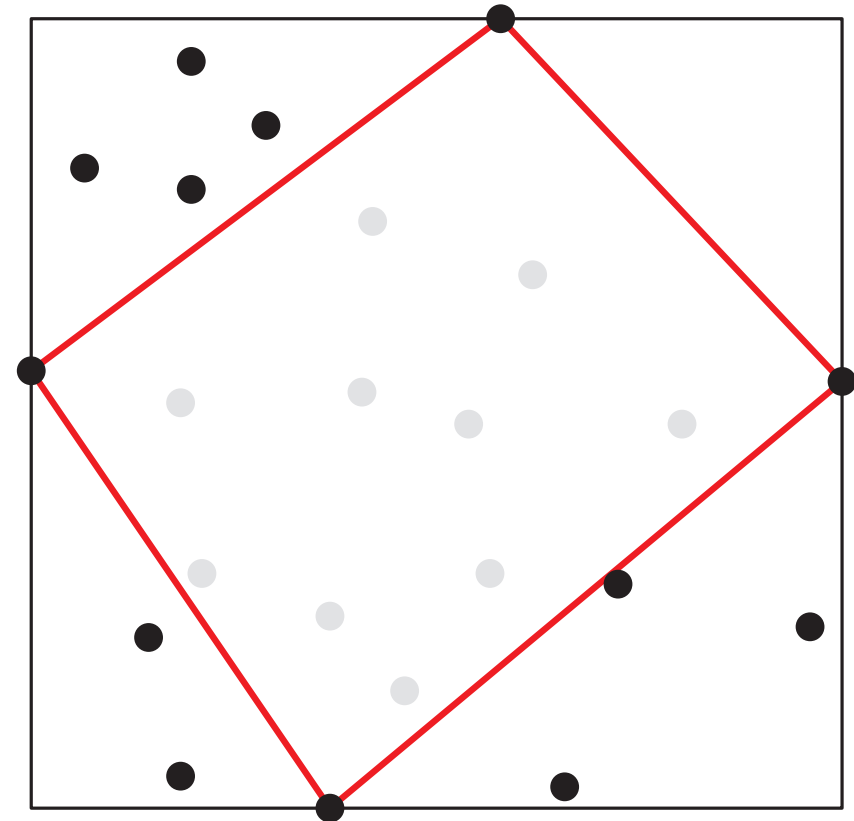
Recursively, do:

1  
2  
3

points lying in each region, find the point in the direction orthogonal to the line that furthest from the line. This point terminates the region.

2. Find the extreme point with the same endpoints of the line and update the convex hull.

3. Repeat the process for the remaining points of each region, and update the convex hull according to their position (left or right) and eliminate them if they lie in the interior of the created triangle.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

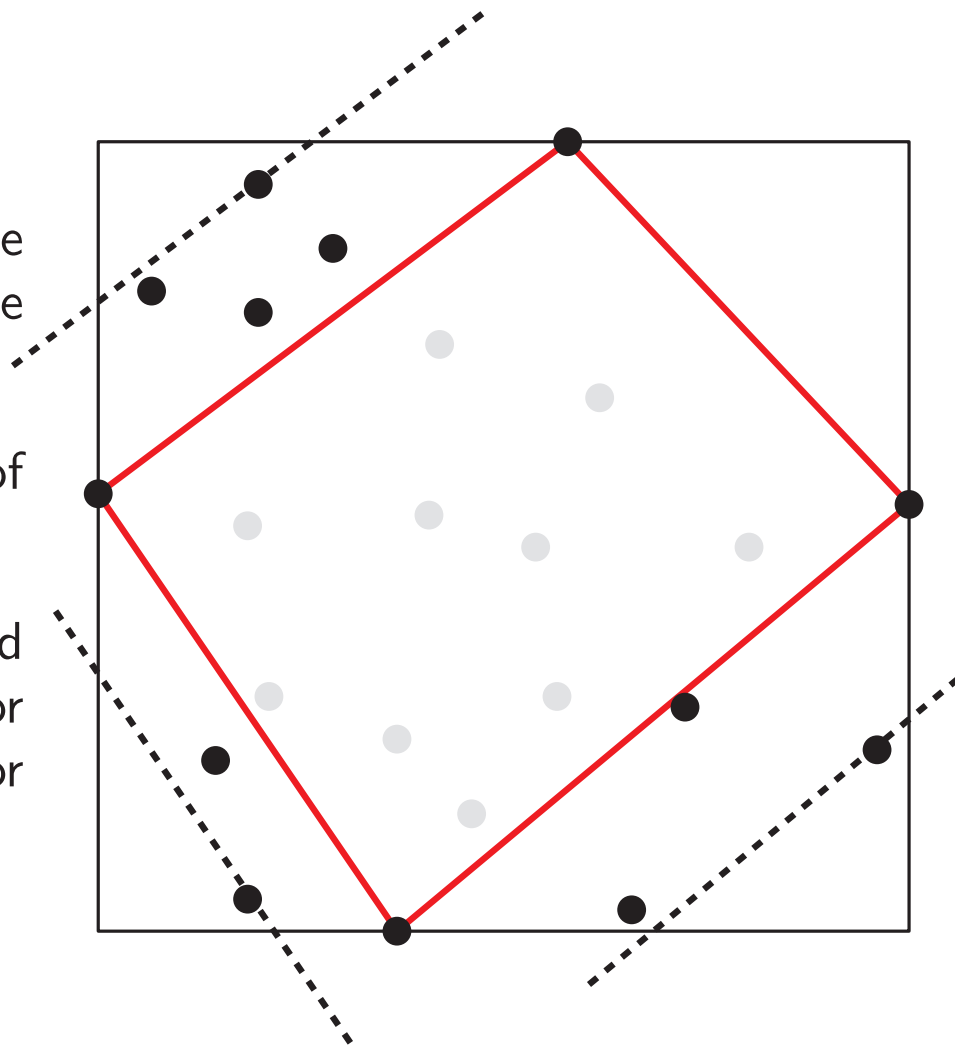
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the line that defines the region.

2. Connect the extreme point with the endpoints of the line and update the convex hull.

3. Repeat for the remaining points of each region, and prune them according to their position (left or right) relative to the line that defines the region created triangle.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

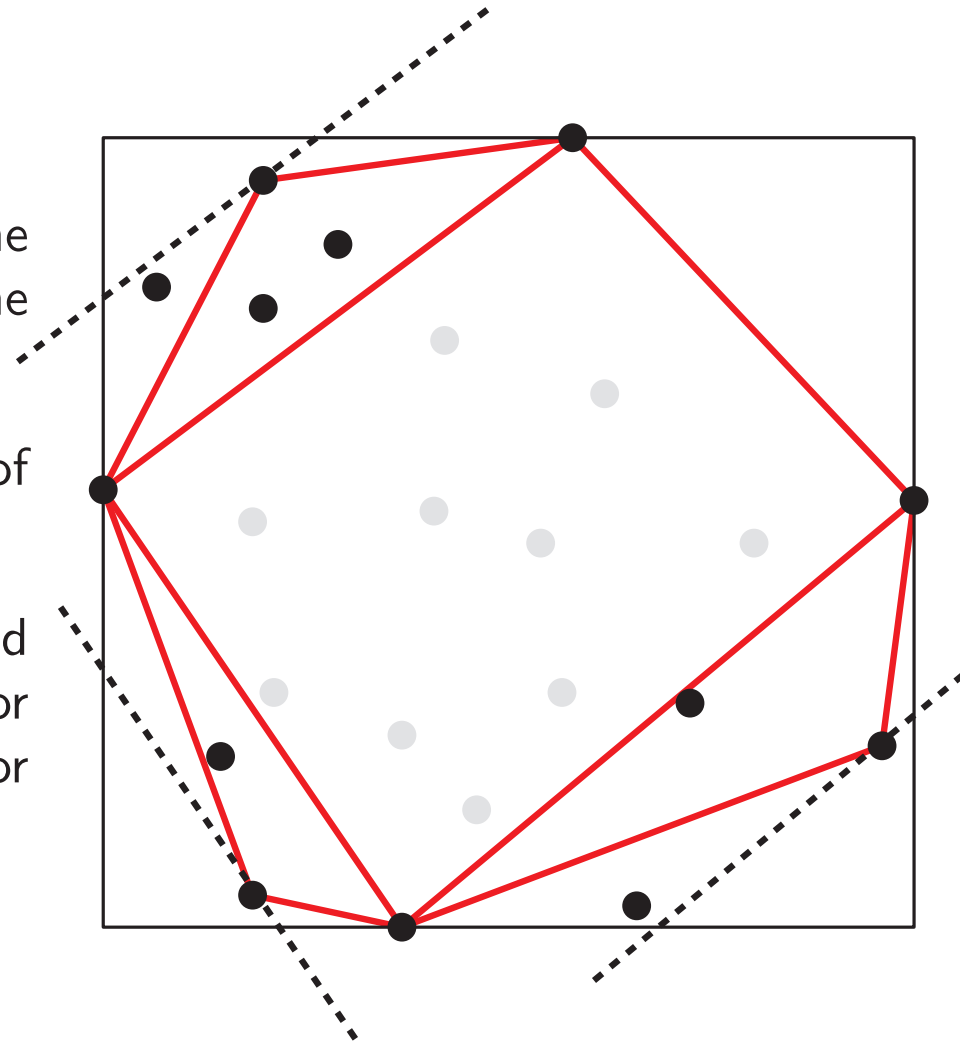
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the line that defines the region.

2. Connect the extreme point with the endpoints of the line and update the convex hull.

3. Repeat for the remaining points of each region, and prune them according to their position (left or right) relative to the line created in the previous step.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

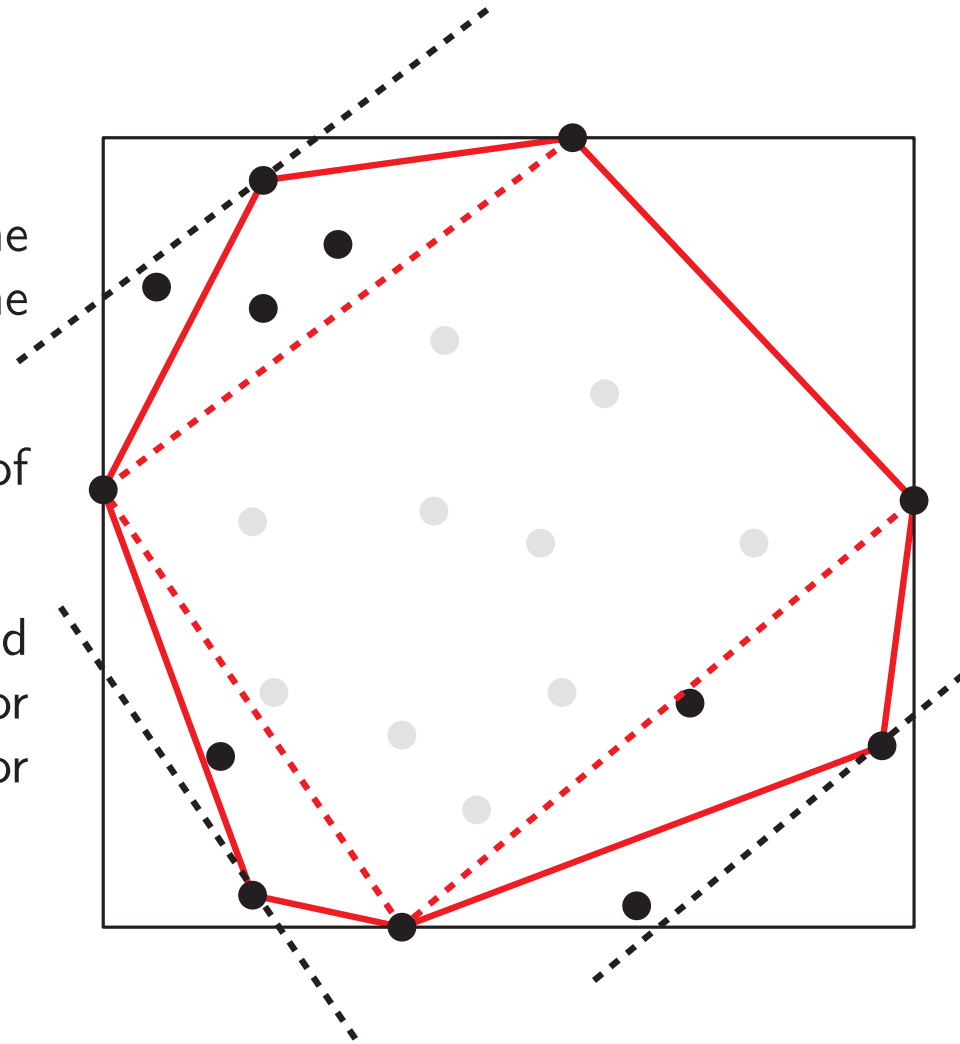
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the line that defines the region.

2. Connect the extreme point with the endpoints of the line and update the convex hull.

3. Repeat for the remaining points of each region, and prune them according to their position (left or right) relative to the line created in the previous step.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

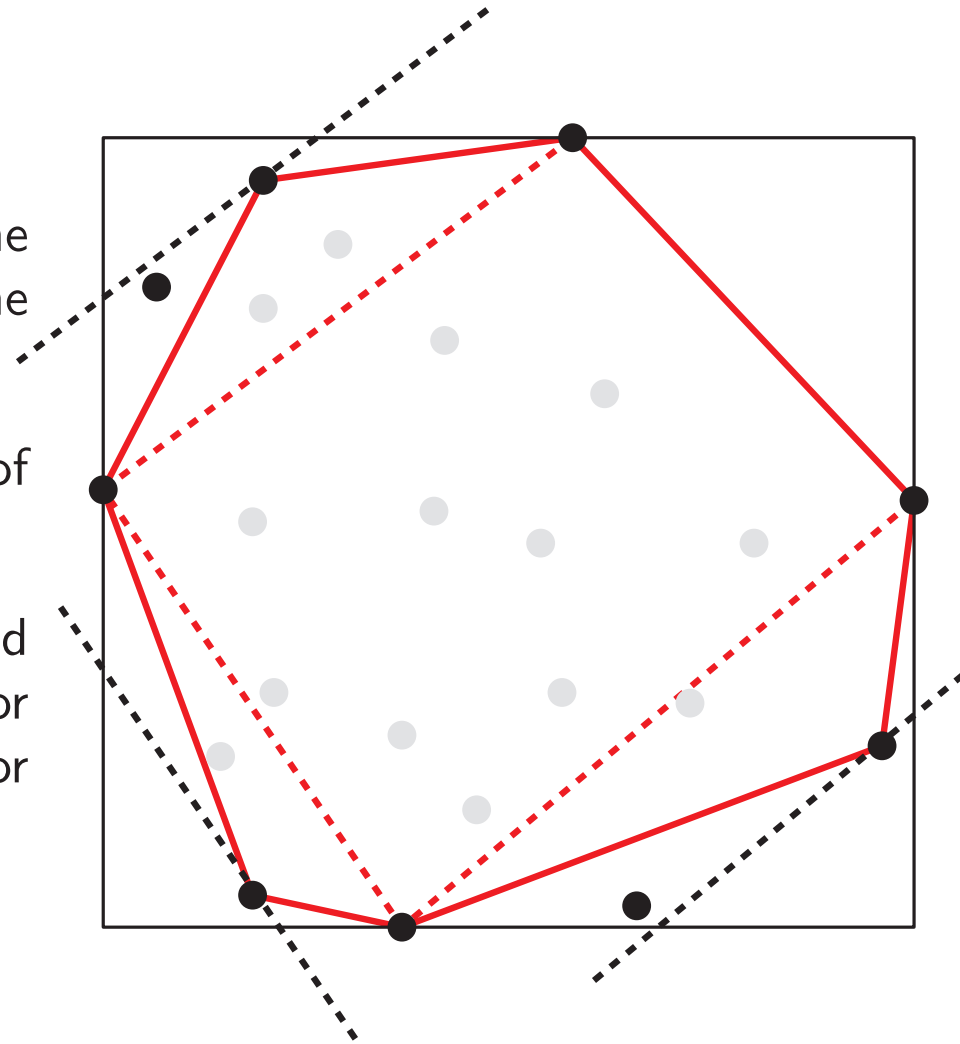
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the line that defines the region.

2. Connect the extreme point with the endpoints of the line and update the convex hull.

3. Repeat for the remaining points of each region, and prune them according to their position (left or right) relative to the line created in the previous step.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

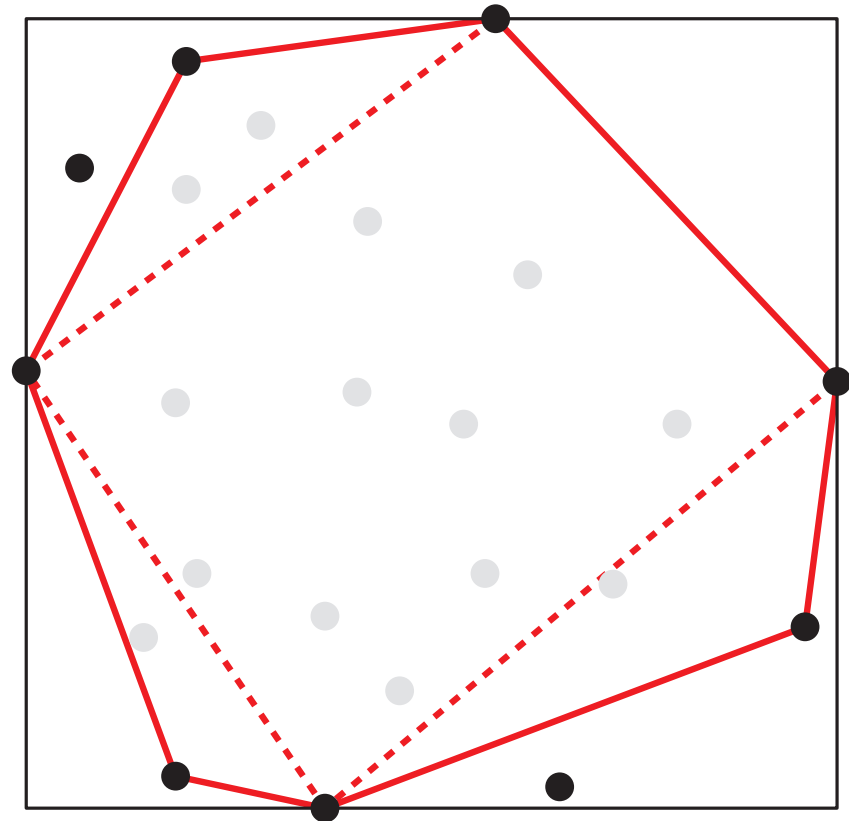
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the line segment that defines the region.

2. For each region, find the extreme point with the same endpoints of the line segment and update the convex hull.

3. For each region, find the extreme point of each region, and update the convex hull according to their position (left or right) and eliminate them if they lie in the interior of the region created triangle.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

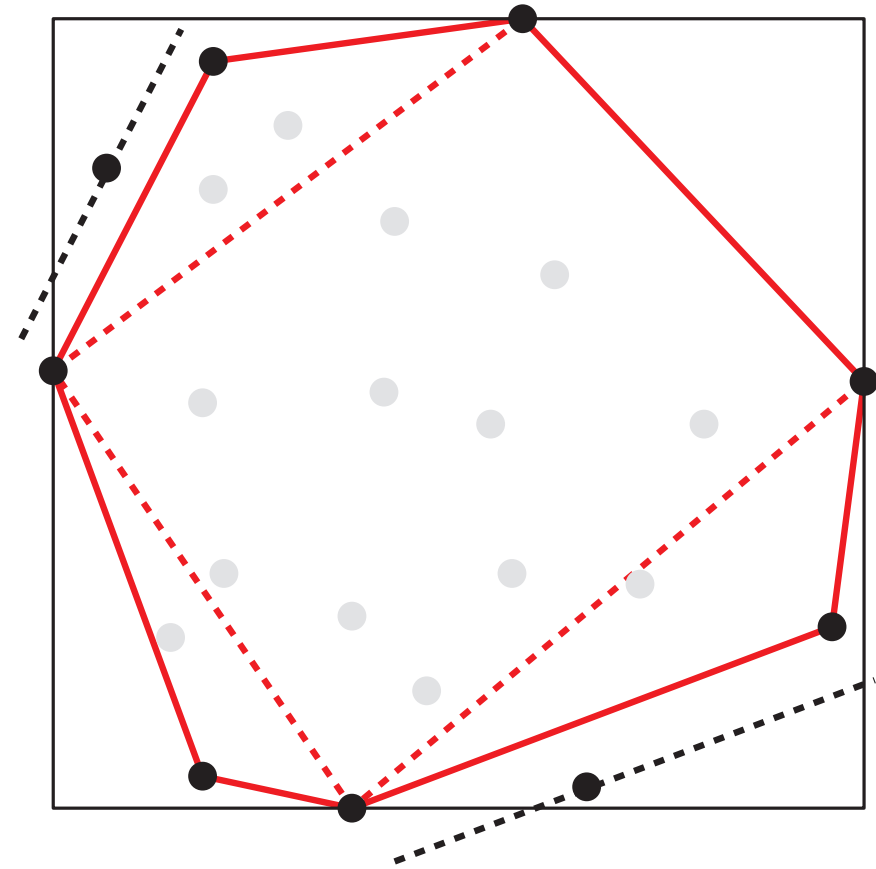
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the current edge. This point terminates the region.

2. For each region, find the extreme point with the same endpoints of the current edge and update the convex hull.

3. For each region, find the extreme point of each region, and update the convex hull according to their position (left or right) and eliminate them if they lie in the interior of the current edge or the created triangle.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

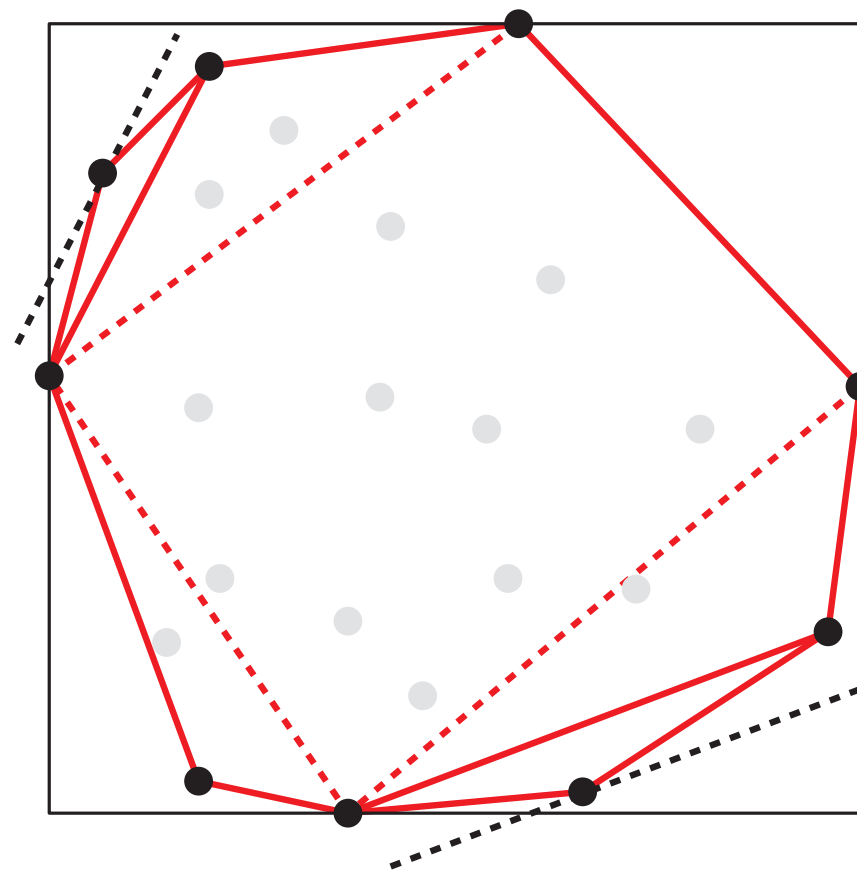
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the current edge. This point terminates the region.

2. For each region, find the extreme point with the same endpoints of the current edge and update the convex hull.

3. For each region, find the extreme point of each region, and update the convex hull according to their position (left or right) and prune them if they lie in the interior of the current edge.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

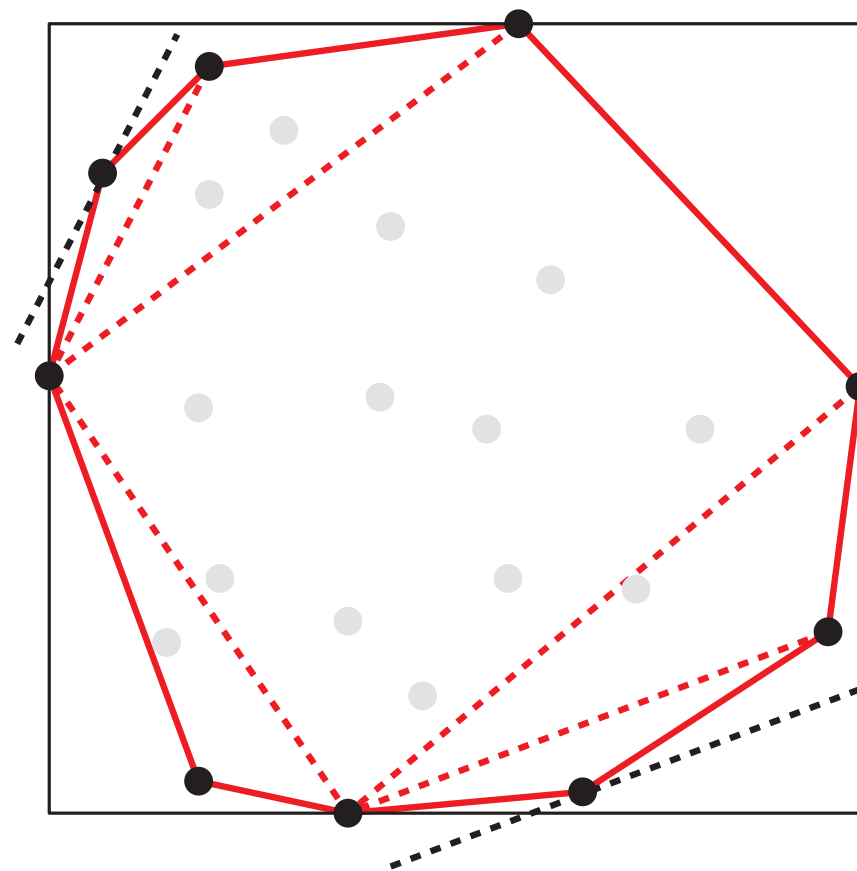
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the edge that terminates the region.

2. For each region, find the extreme point with the endpoints of the edge and update the convex hull.

3. For each region, find the remaining points of each region, and prune them according to their position (left or right) relative to the edge of the created triangle.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

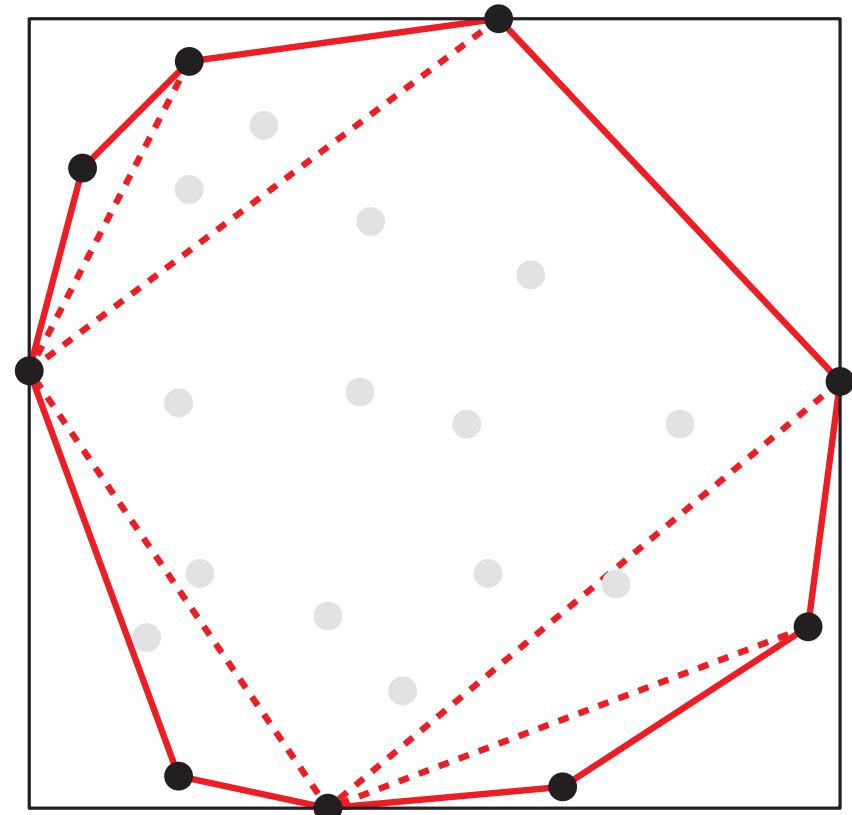
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the line segment that defines the region.

2. For each region, find the extreme point with the same endpoints of the line segment and update the convex hull.

3. For each region, find the extreme point of each region, and update the convex hull according to their position (left or right) and prune them if they lie in the interior of the region created triangle.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

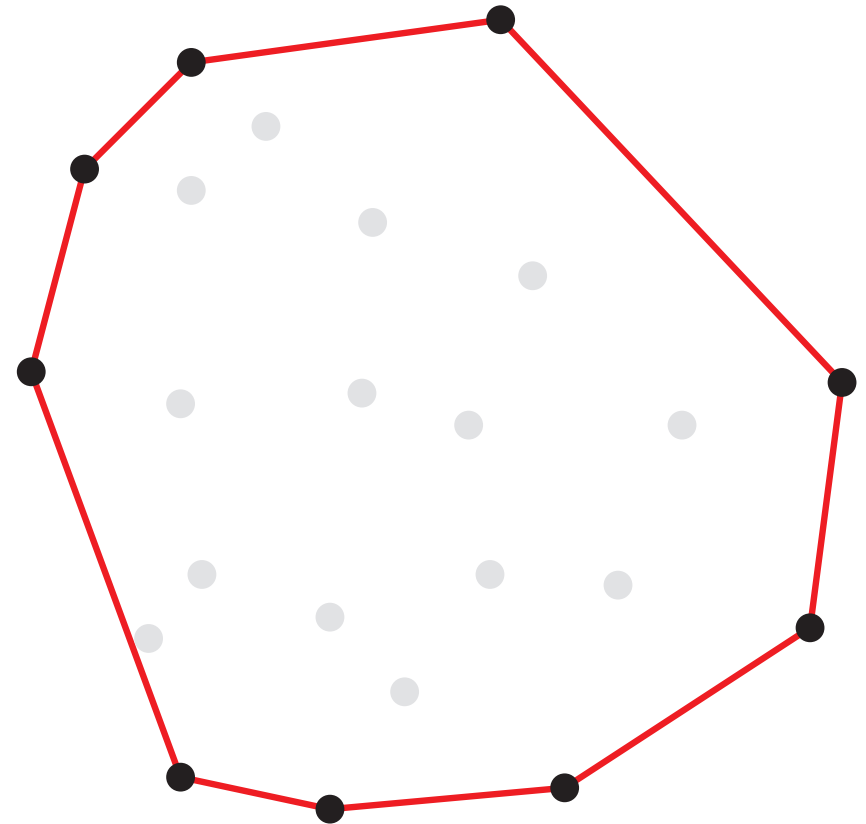
Recursively, do:

1  
2  
3

1. For each region, find the extreme point in the direction orthogonal to the line segment that defines the region.

2. Update the convex hull with the extreme point and update the convex hull.

3. Recursively process the remaining points of each region, and update the convex hull according to their position (left or right) relative to the line segment that defines the region. Prune or eliminate them if they lie in the interior of the region created triangle.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Advance

Recursively, do:

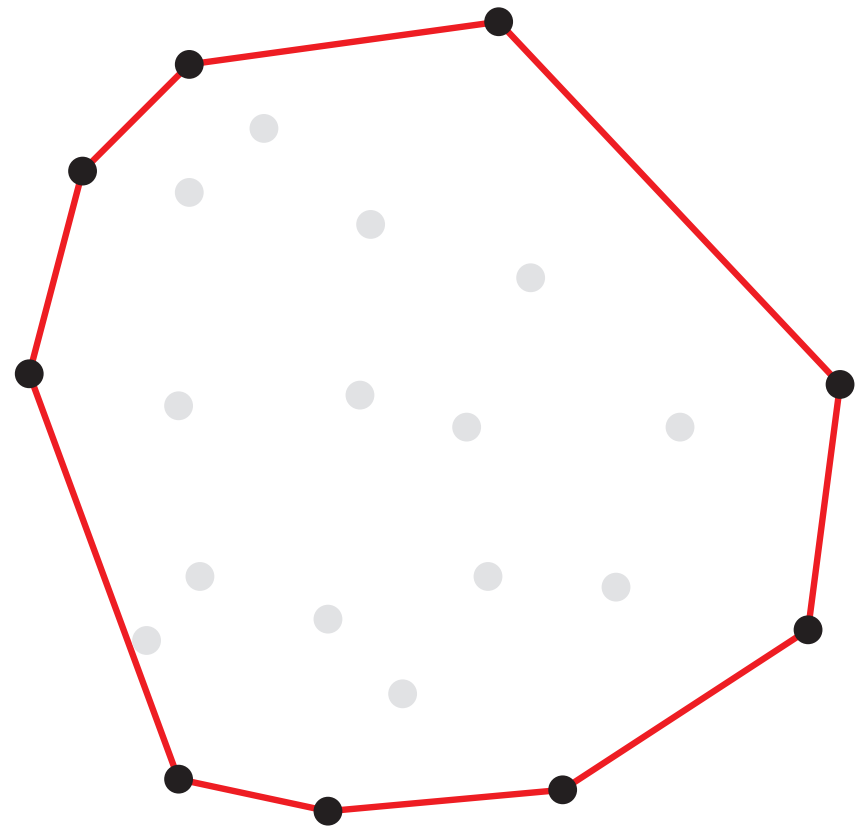
1  
2  
3

points lying in each region, find the point in the direction orthogonal to the line that furthest from the line and terminates the region.

Find the extreme point with the endpoints of the line and update the convex hull.

Repeat the process on the remaining points of each region, and update the convex hull according to their position (left or right) and eliminate them if they lie in the interior of the region created triangle.

Run this step:  $O(n^2)$



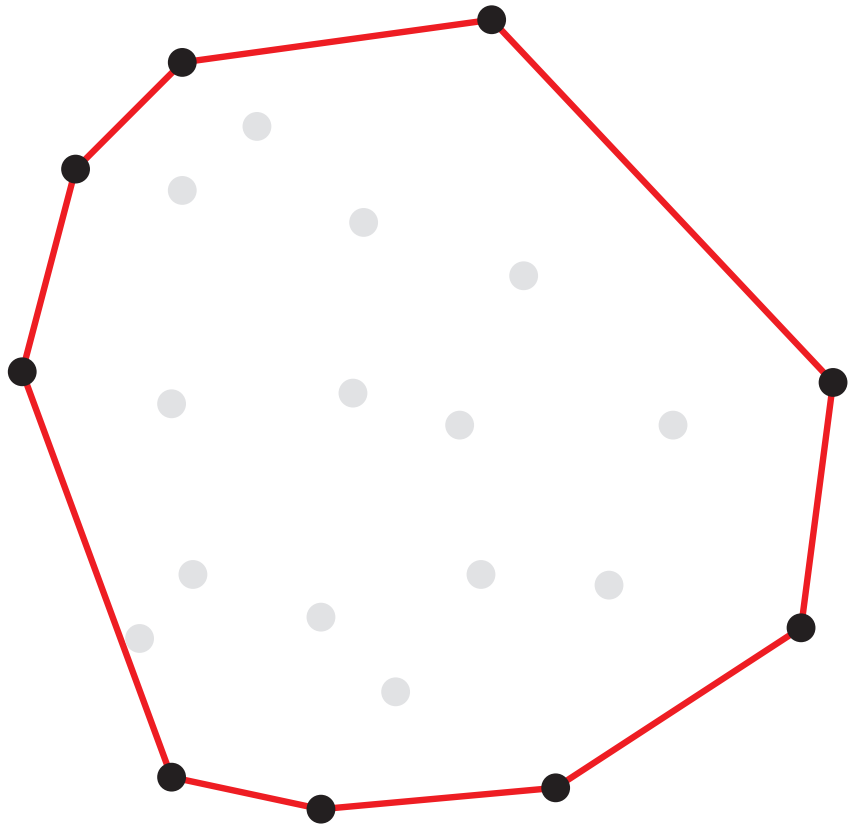
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Run

# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

Overall running time:  $O(n^2)$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
- - -  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# CONVEX HULL IN 2D

## QuickHull algorithm (by prune-and-search)

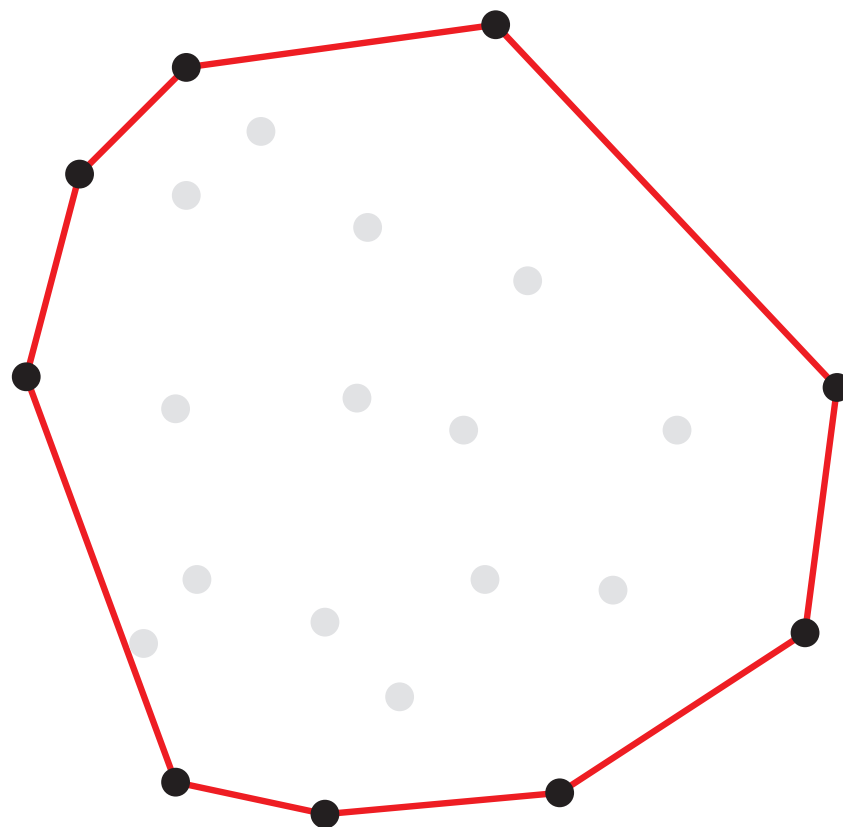
Overall running time:  $O(n^2)$

running time of this algorithm depends on the distribution of the input points.

If all input points are in convex position, the running time is  $\Theta(n^2)$ .

If the input points are such that each prune step eliminates a constant fraction of the current points, then the algorithm runs in  $\Theta(n \log n)$  time.

If the input hull is triangular, the algorithm runs in  $\Theta(n)$  time.



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Graham's algorithm

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

--

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

...  $s(\text{top}(l))$

... left turn:

... in  $l$

...  $e_i$

... from  $l$

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
 ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
 CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al  
 Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002.  
 Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the first point in  $l$  and delete it from  $P$

At each point  $p_i \in P$  to be explored, do:

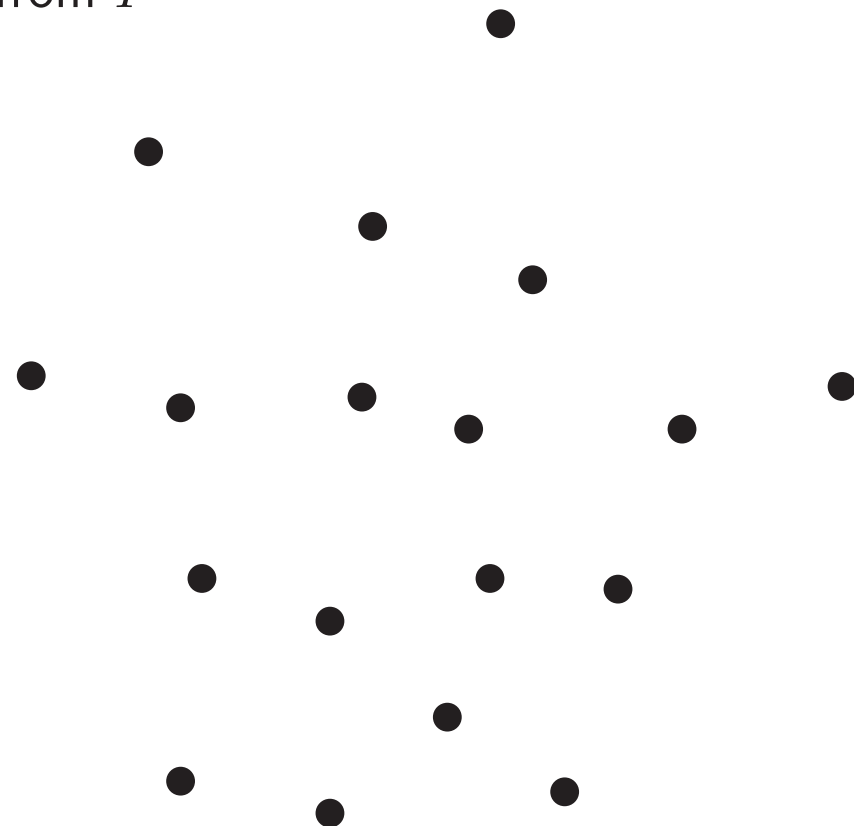
$s = s(top(l))$

is a left turn:

push  $p_i$  in  $l$

else

pop from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenap9.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

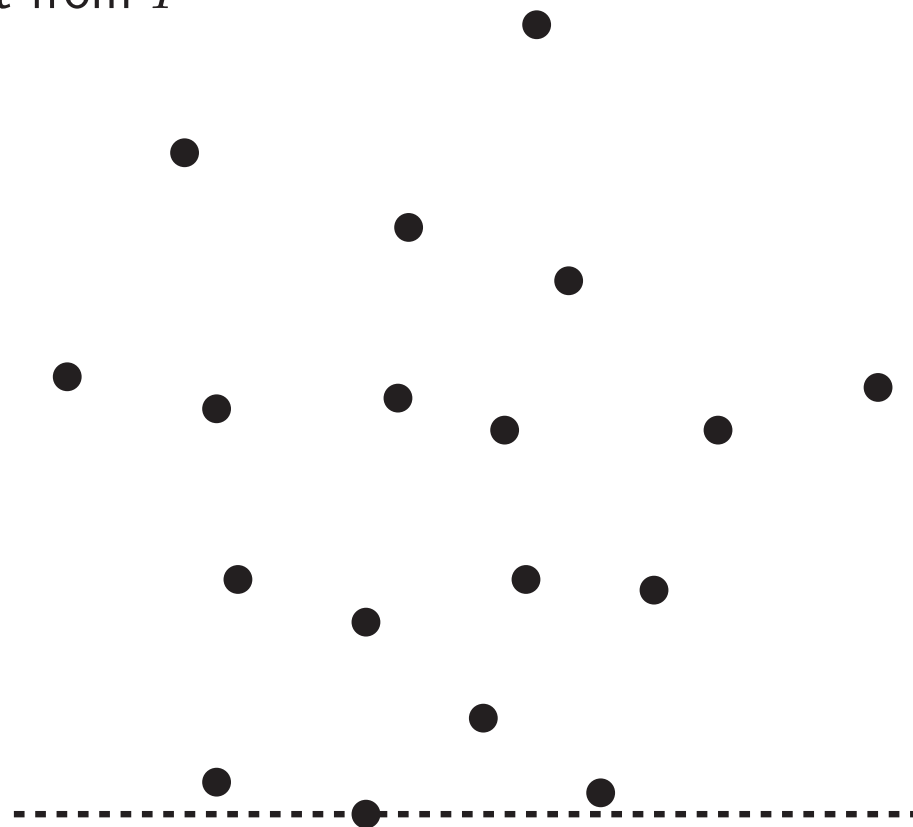
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it from  $P$

At each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

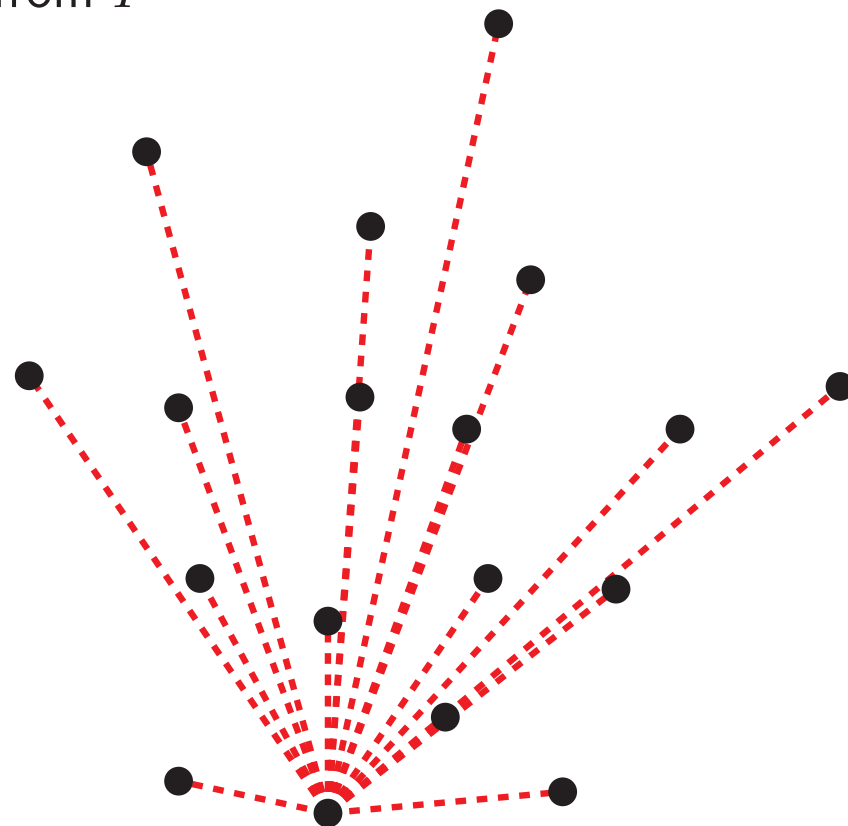
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it from  $P$

At each point  $p_i \in P$  to be explored, do:

- If  $\angle(p_{i-1}, p_i, p_{i-2})$  is a left turn:
  - Push  $p_i$  in  $l$
  - Delete  $p_{i-1}$  from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

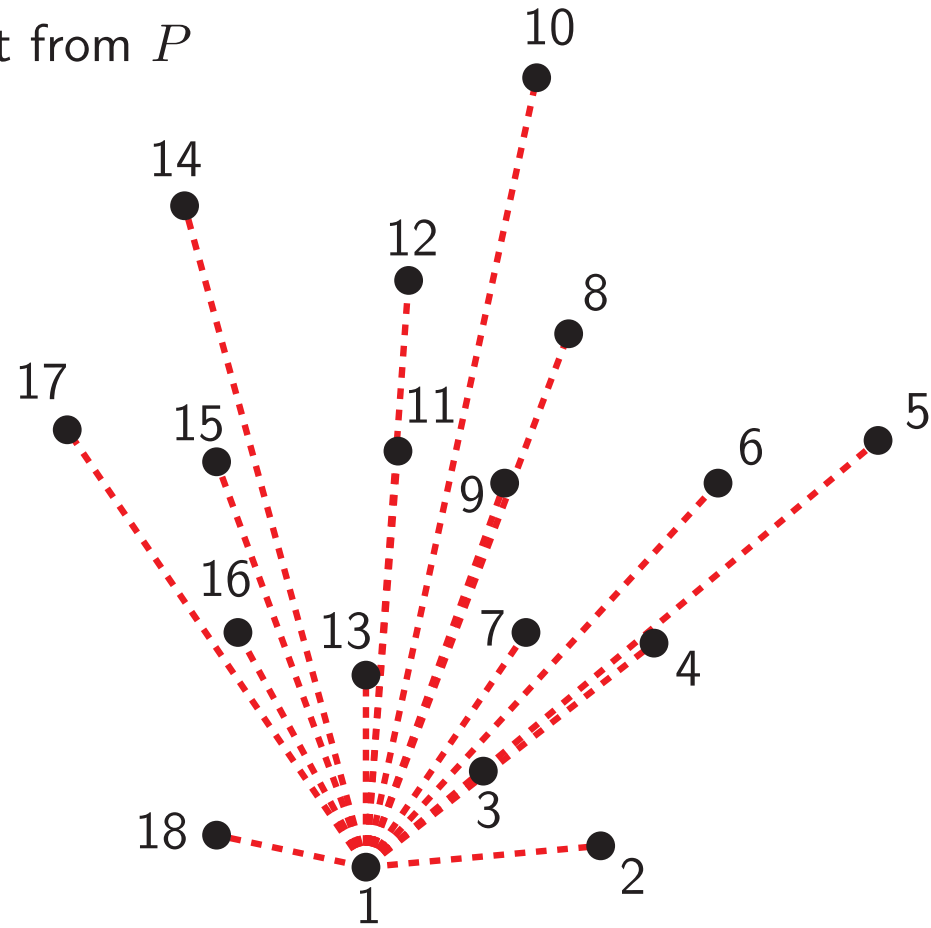
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $\angle(p_i, \text{top}(l))$  is a left turn:
  - Push  $p_i$  in  $l$
  - Else if  $p_i$  is on the line defined by  $\text{top}(l)$  and  $\text{second}(l)$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

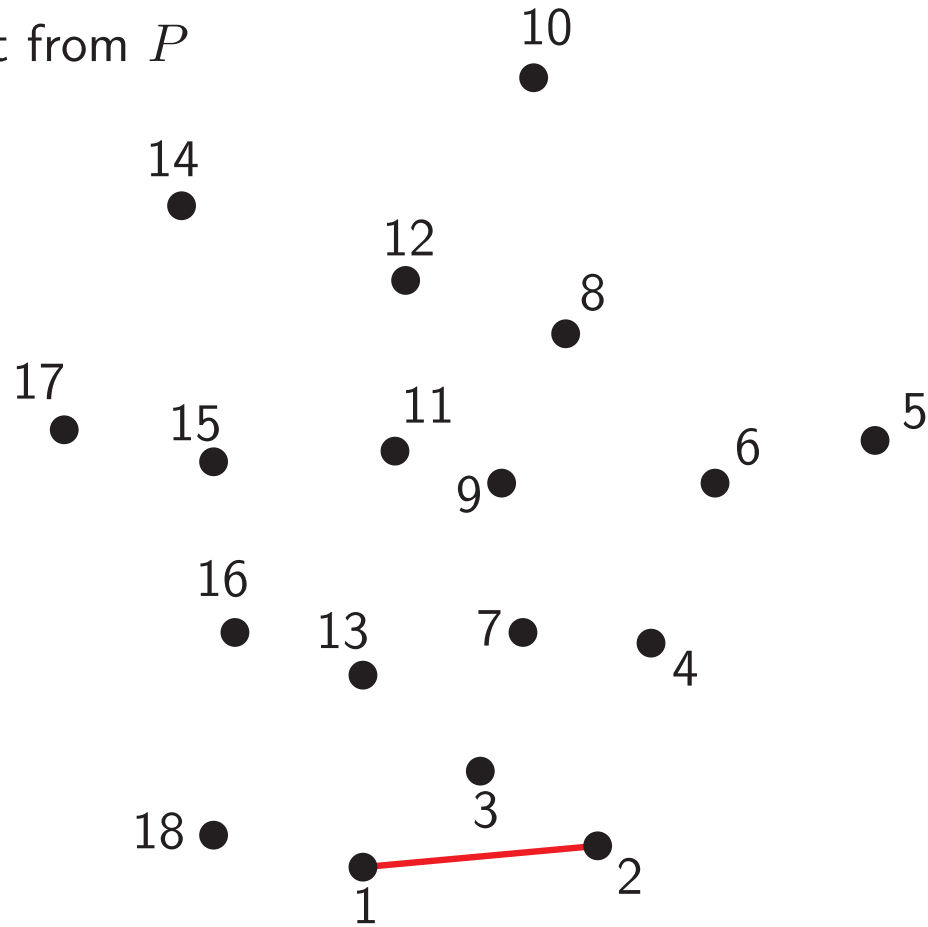
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it from  $P$

For each point  $p_i \in P$  to be explored, do:

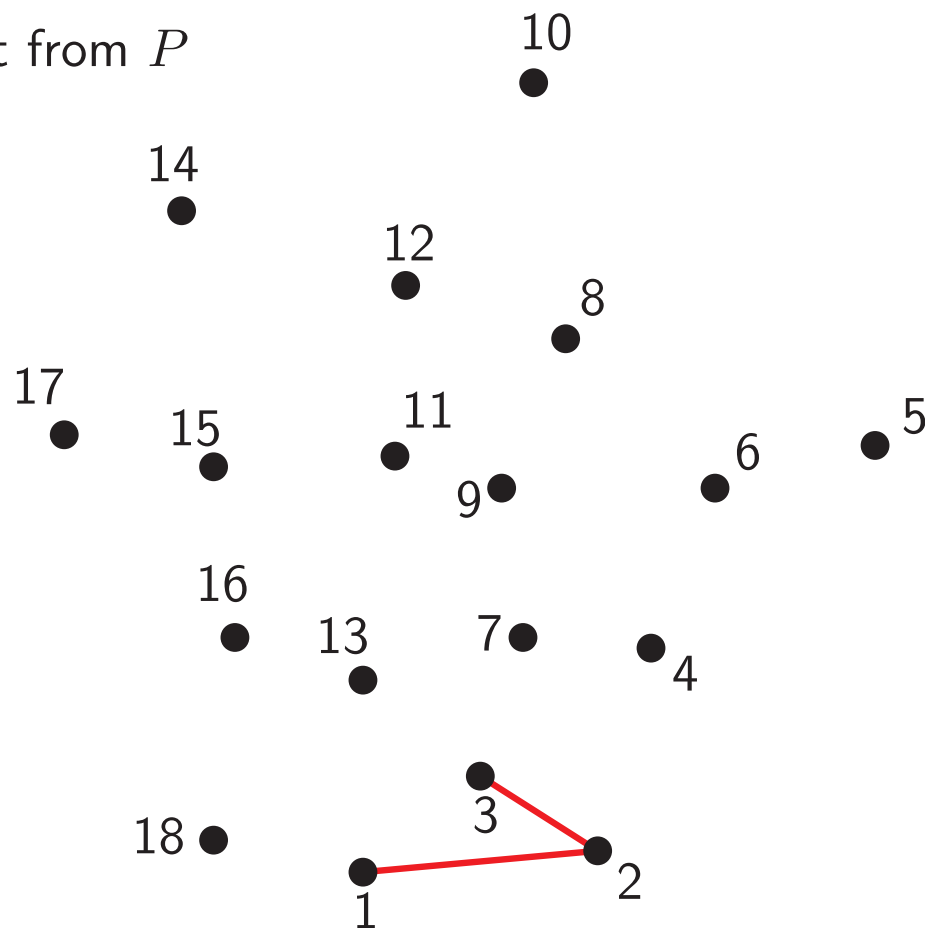
if  $p_i$  is to the left of  $l$ :

push  $p_i$  in  $l$

else if  $p_i$  is on  $l$ :

delete  $p_i$

pop from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it from  $P$

For each point  $p_i \in P$  to be explored, do:

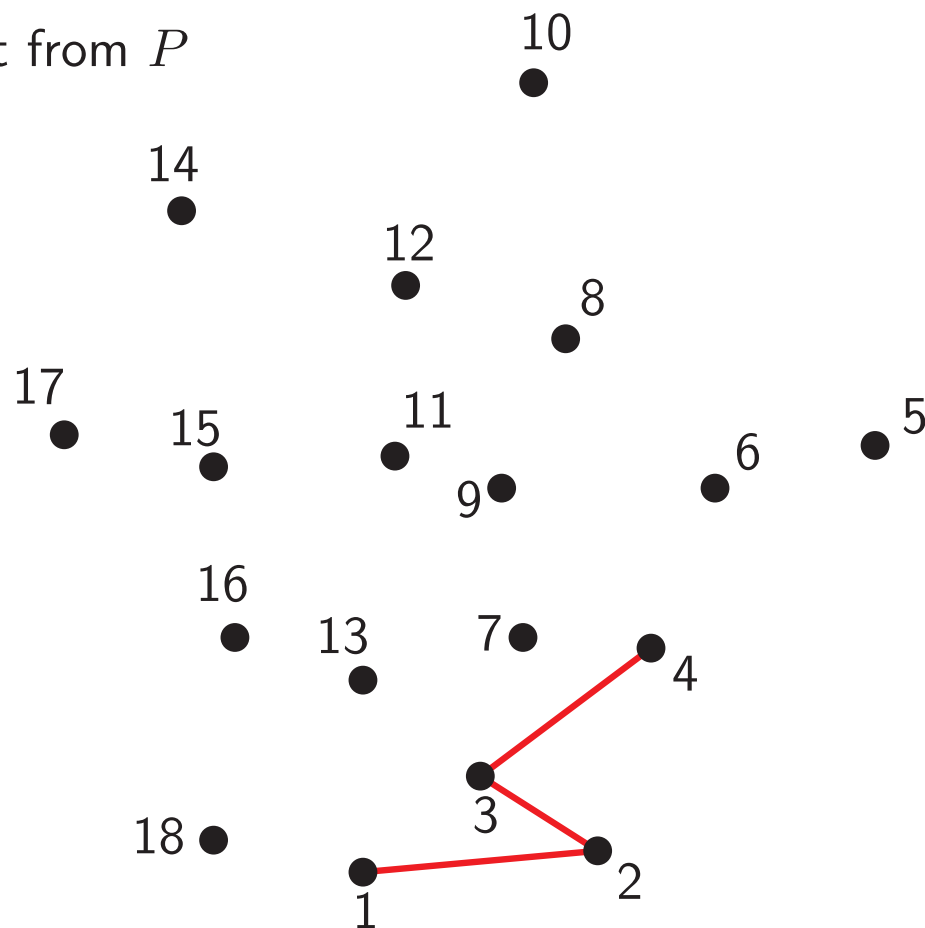
if  $p_i$  is to the left of  $l$ :

push  $p_i$  in  $l$

else if  $p_i$  is on  $l$ :

delete  $p_i$  from  $l$

return  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenap9.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

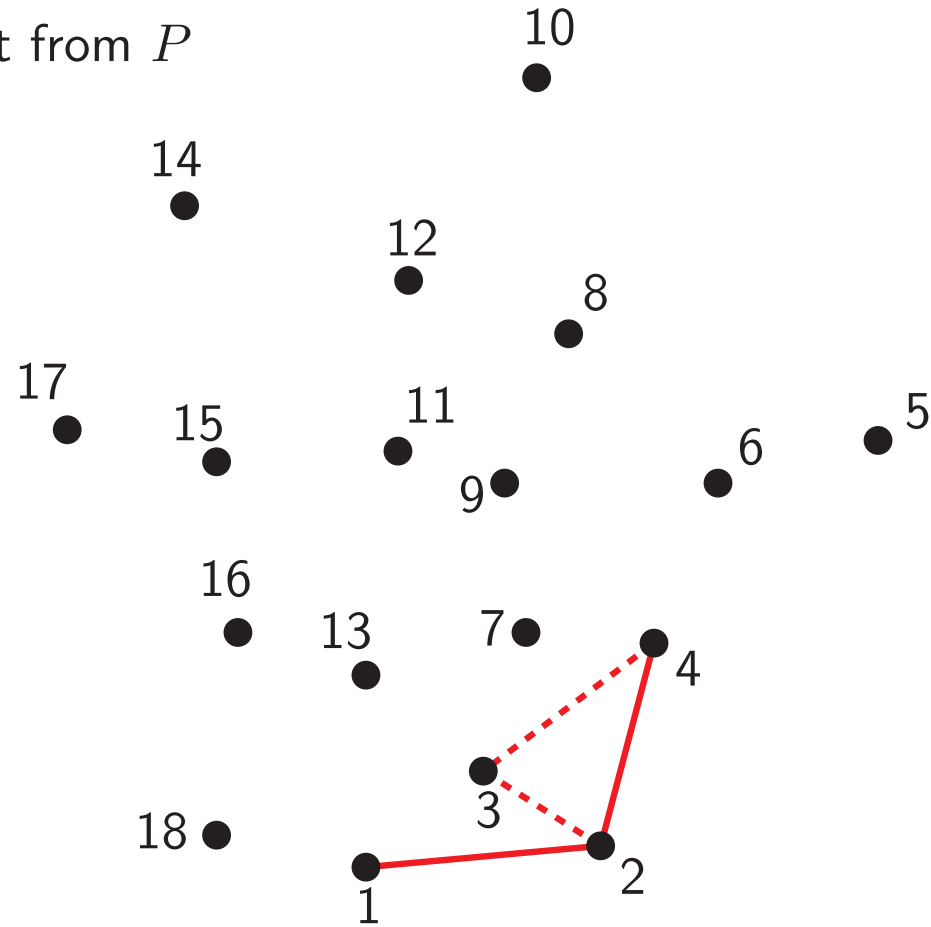
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it from  $P$

For each point  $p_i \in P$  to be explored, do:

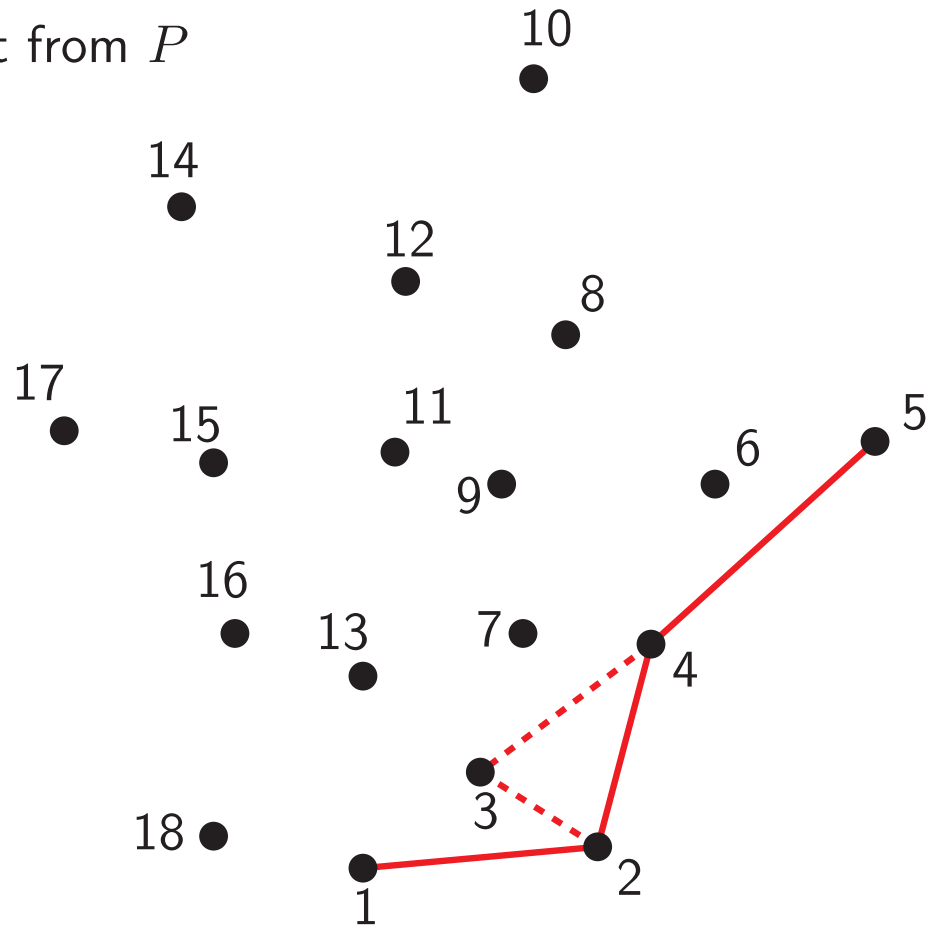
if  $\angle(p_i, \text{top}(l))$

is a left turn:

push  $p_i$  in  $l$

else

pop from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenap9.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it from  $P$

For each point  $p_i \in P$  to be explored, do:

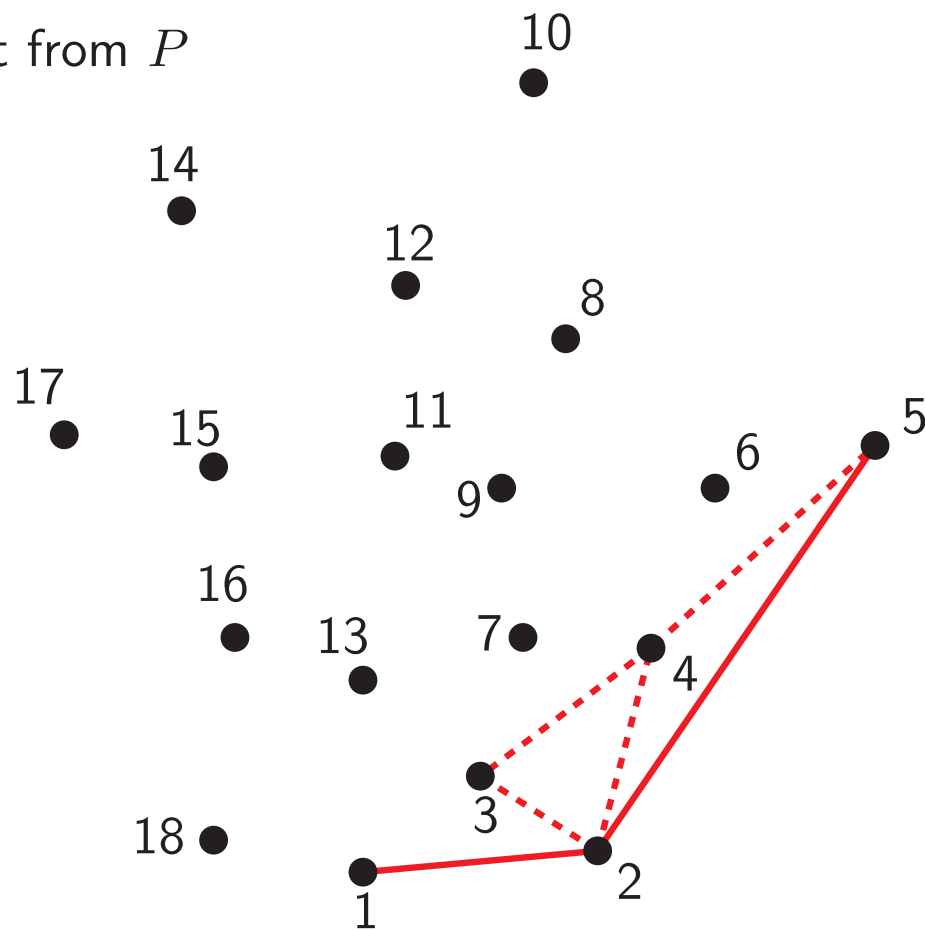
Let  $s = \text{top}(l)$

If a left turn:

Push  $p_i$  in  $l$

else

Pop from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it from  $P$

For each point  $p_i \in P$  to be explored, do:

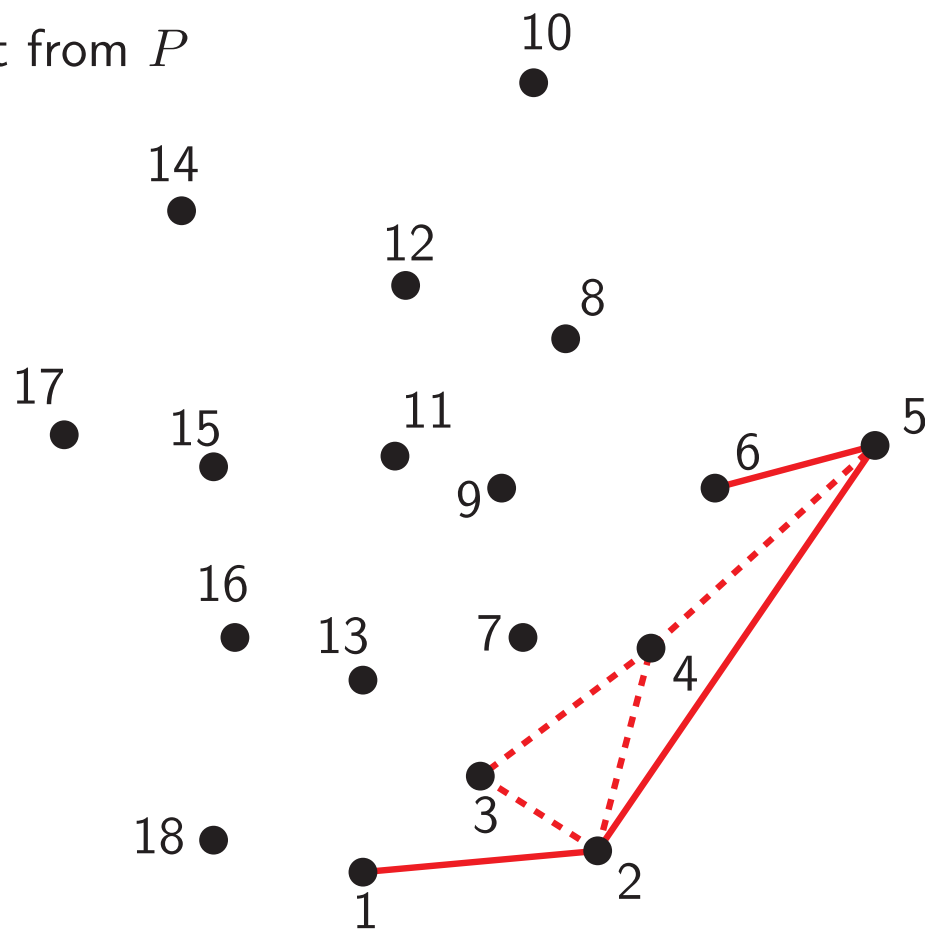
if  $p_i$  is to the left of  $l$ :

if left turn:

push  $p_i$  in  $l$

else:

pop from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

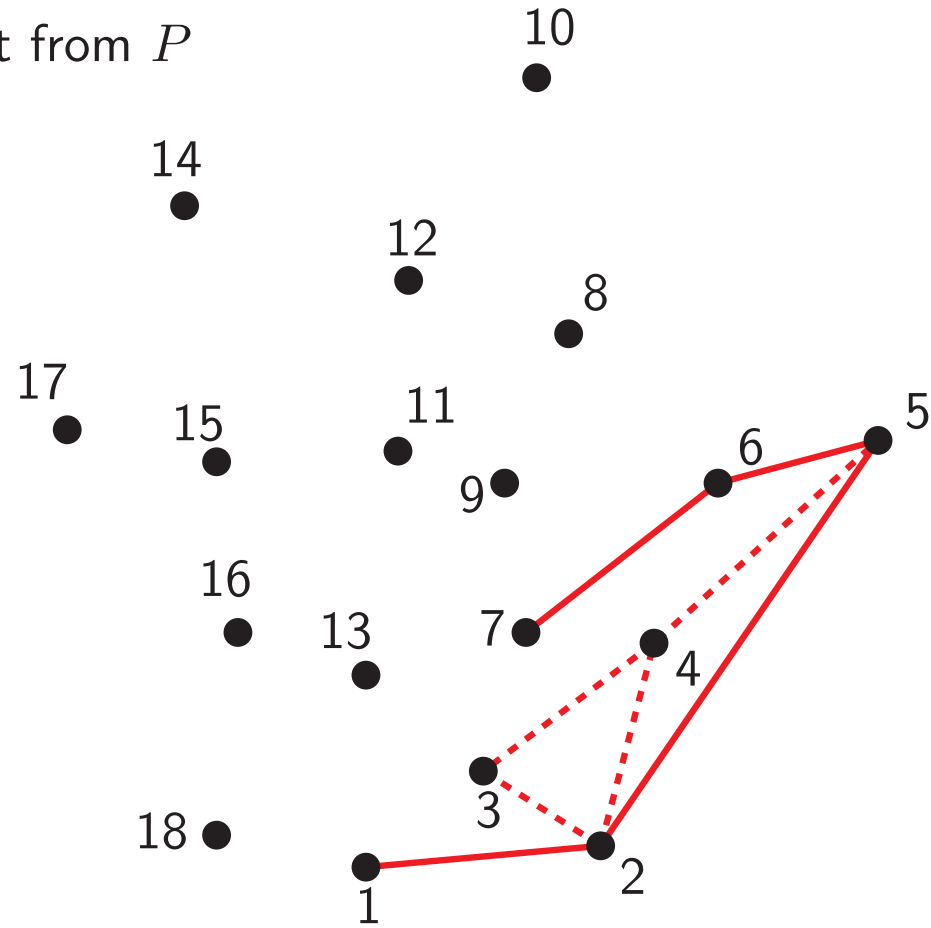
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete if from  $P$

At each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $p_{i-1}$  and  $p_{i-2}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $p_{i-1}$  and  $p_{i-2}$ , delete it from  $l$
- Else if  $p_i$  is to the right of the line through  $p_{i-1}$  and  $p_{i-2}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

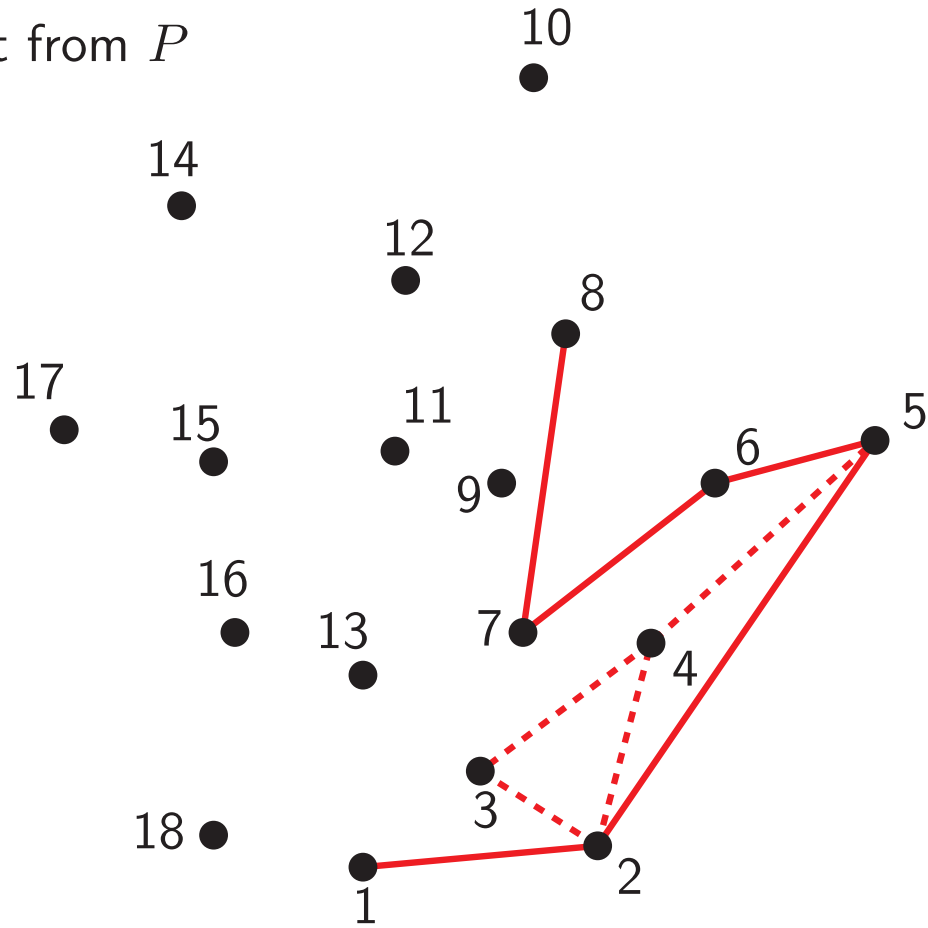
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$
- Else if  $p_i$  is to the right of the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.



# CONVEX HULL IN 2D

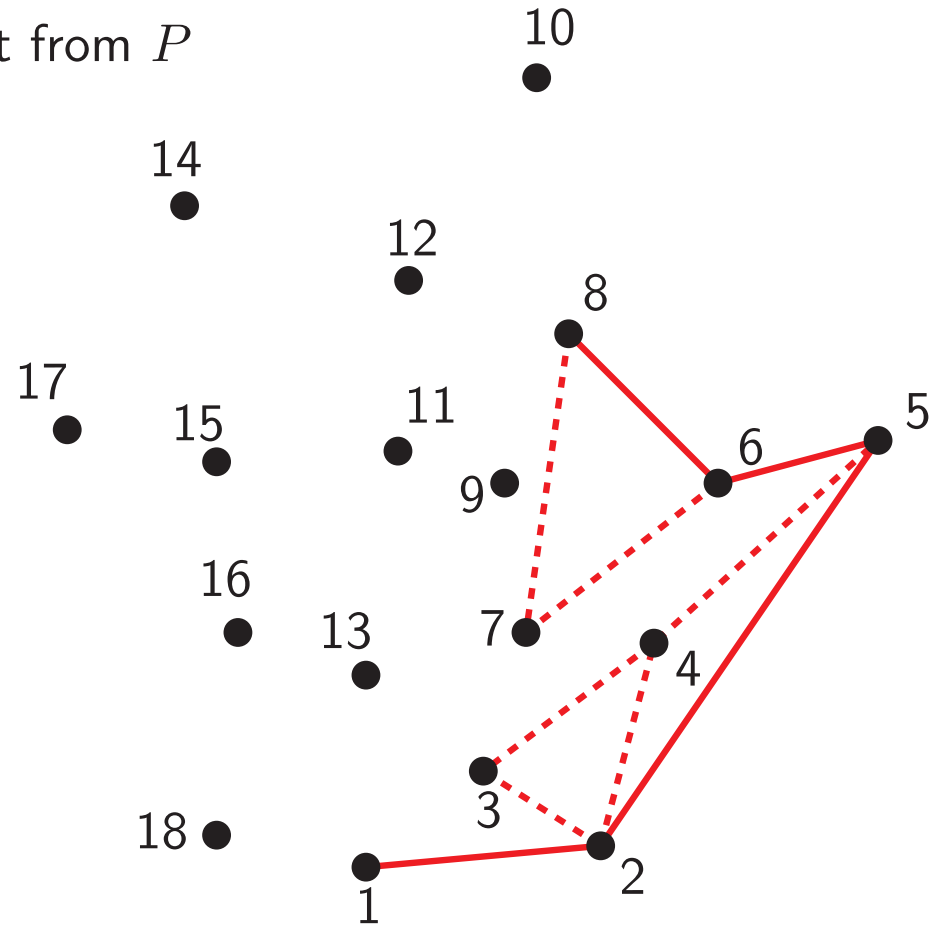
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete it from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$
- Else if  $p_i$  is to the right of the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

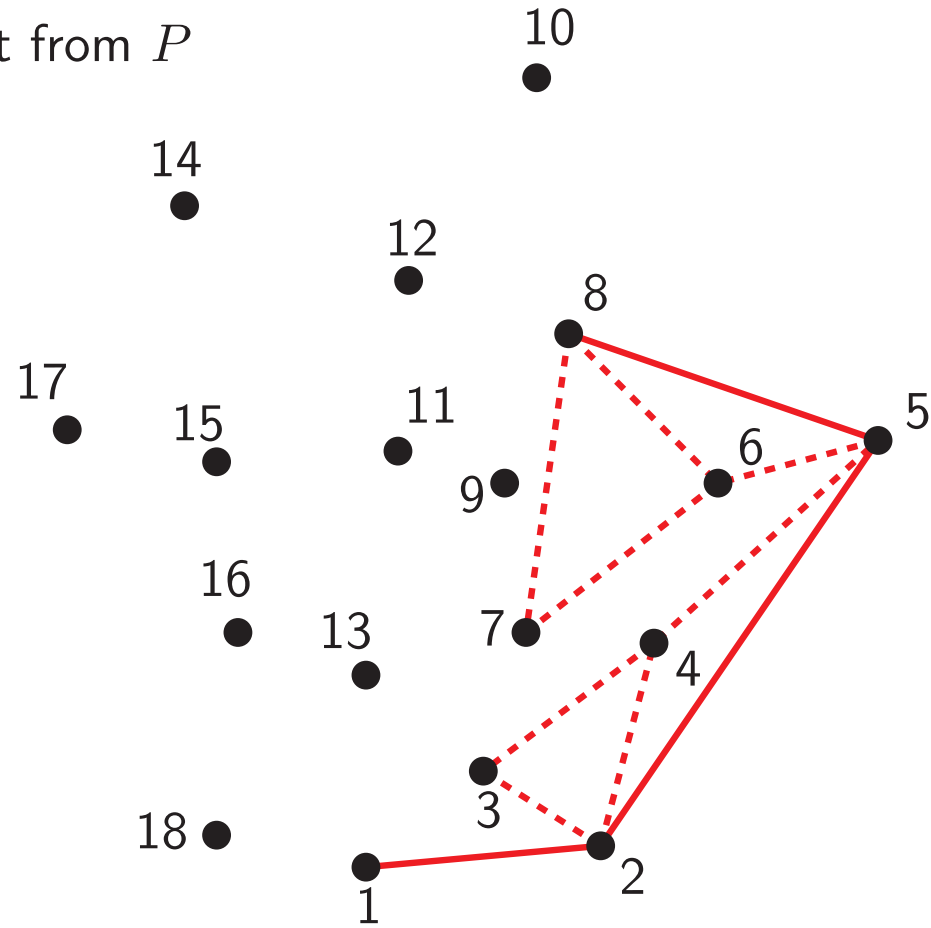
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

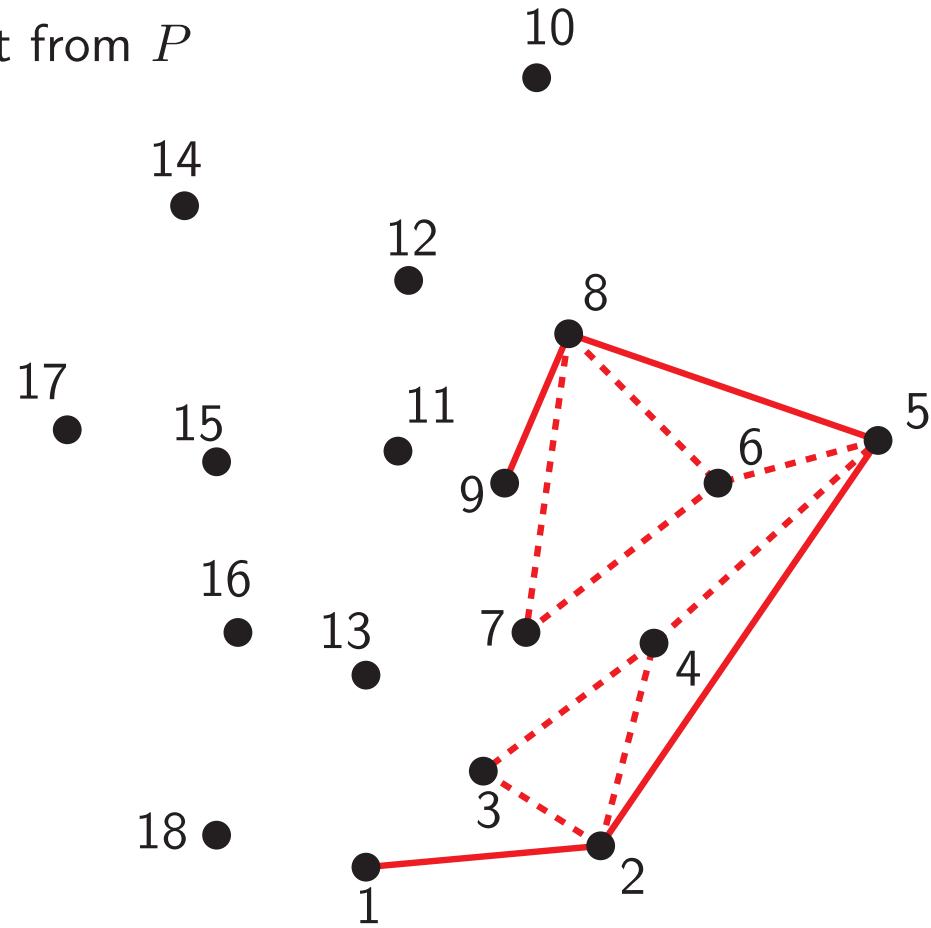
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the next point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$
- Else if  $p_i$  is to the right of the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

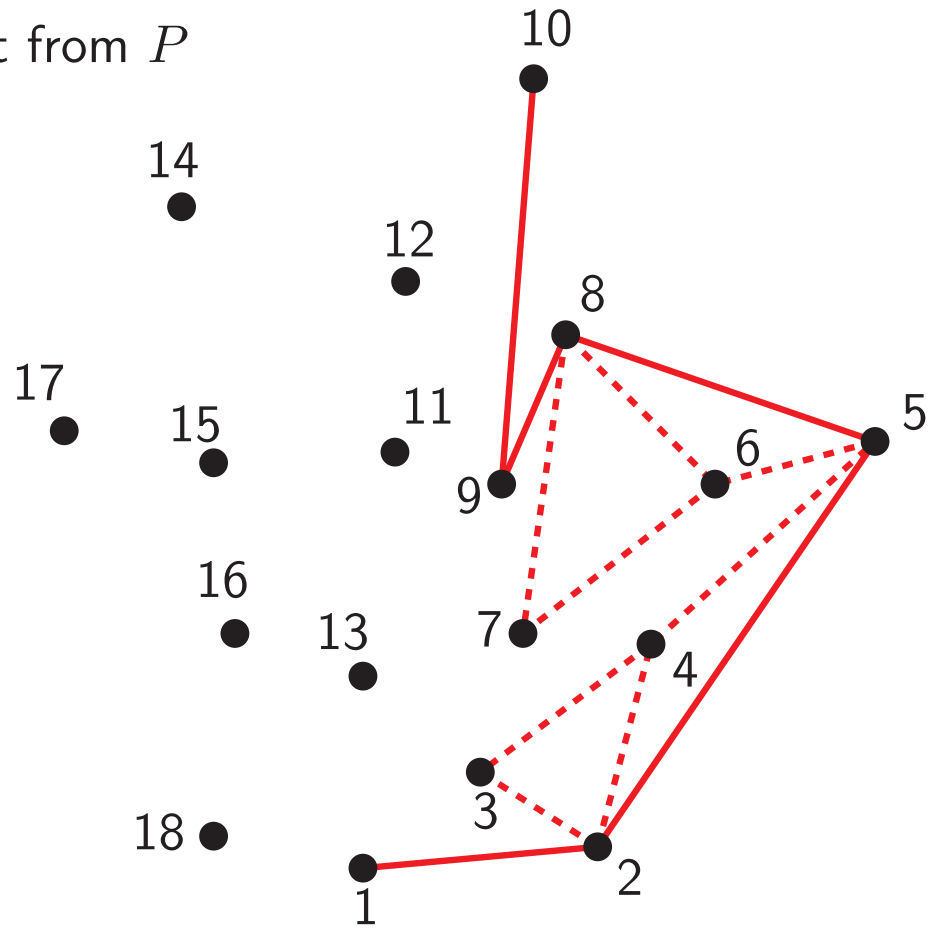
if  $\angle(p_i, \text{top}(l))$

is a left turn:

push  $p_i$  in  $l$

else

pop from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

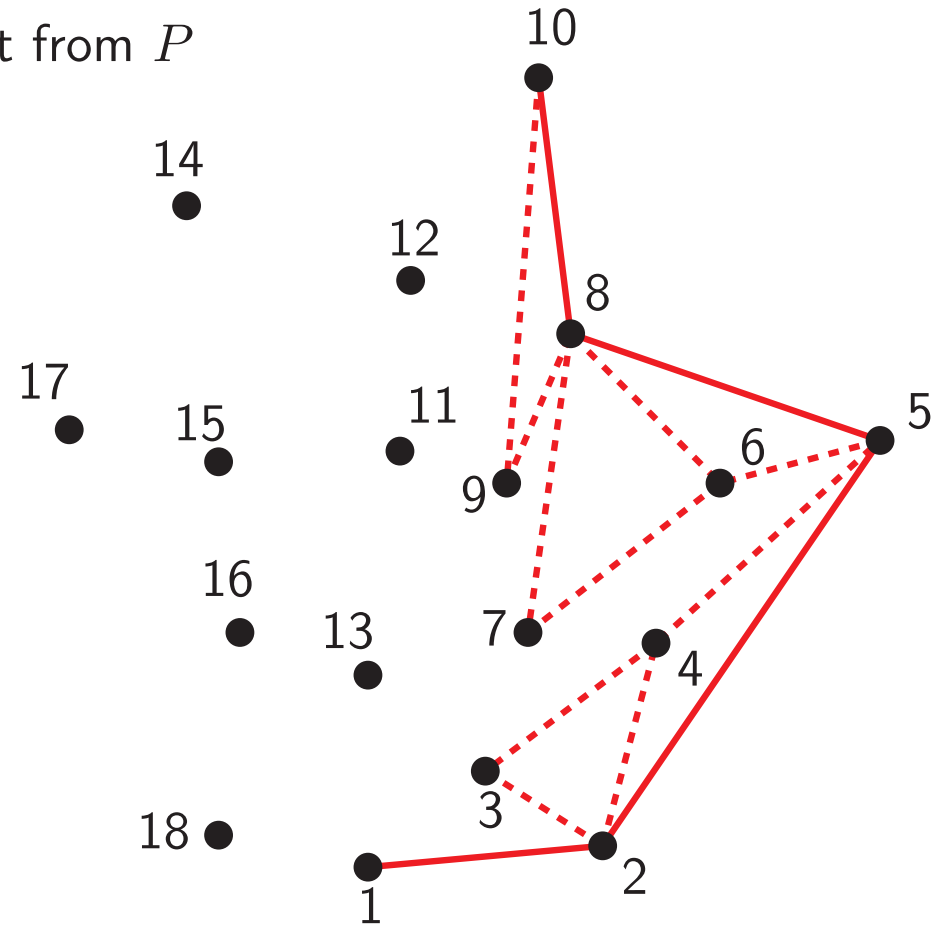
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenap9.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Push the first point in  $l$  and delete it from  $P$

For each point  $p_i \in P$  to be explored, do:

if  $\angle(p_i, \text{top}(l))$

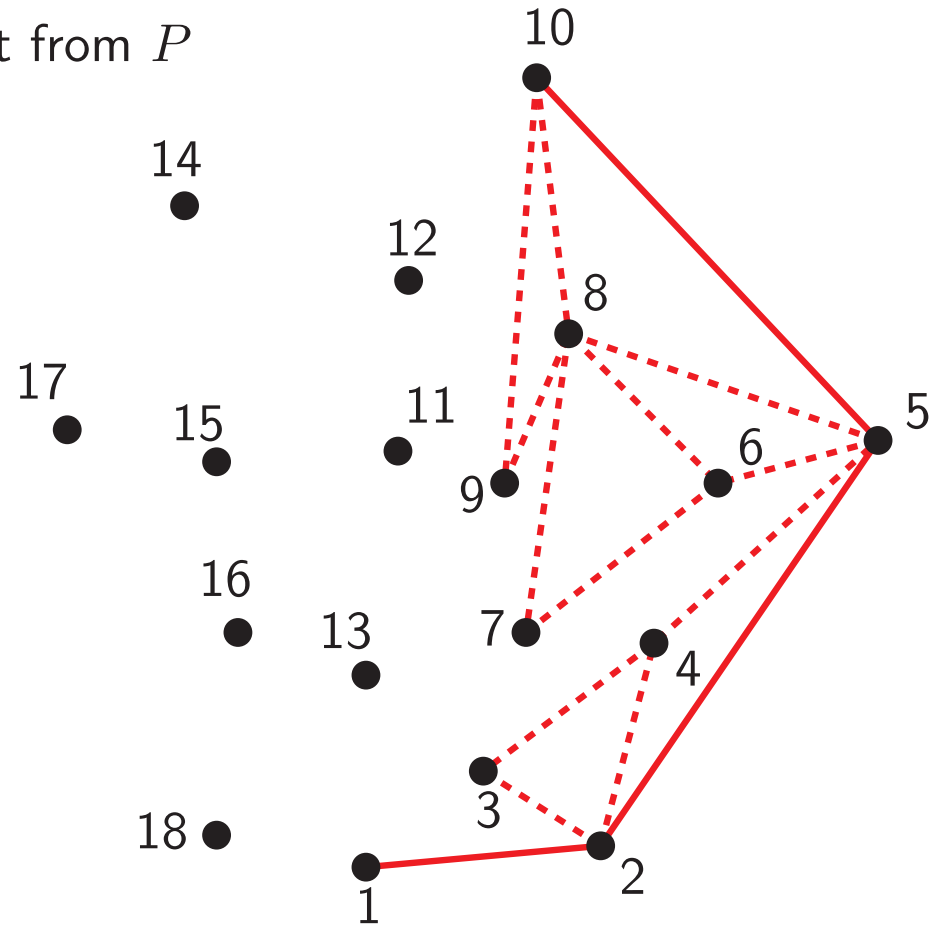
is a left turn:

push  $p_i$  in  $l$

else

pop  $p_i$

pop from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

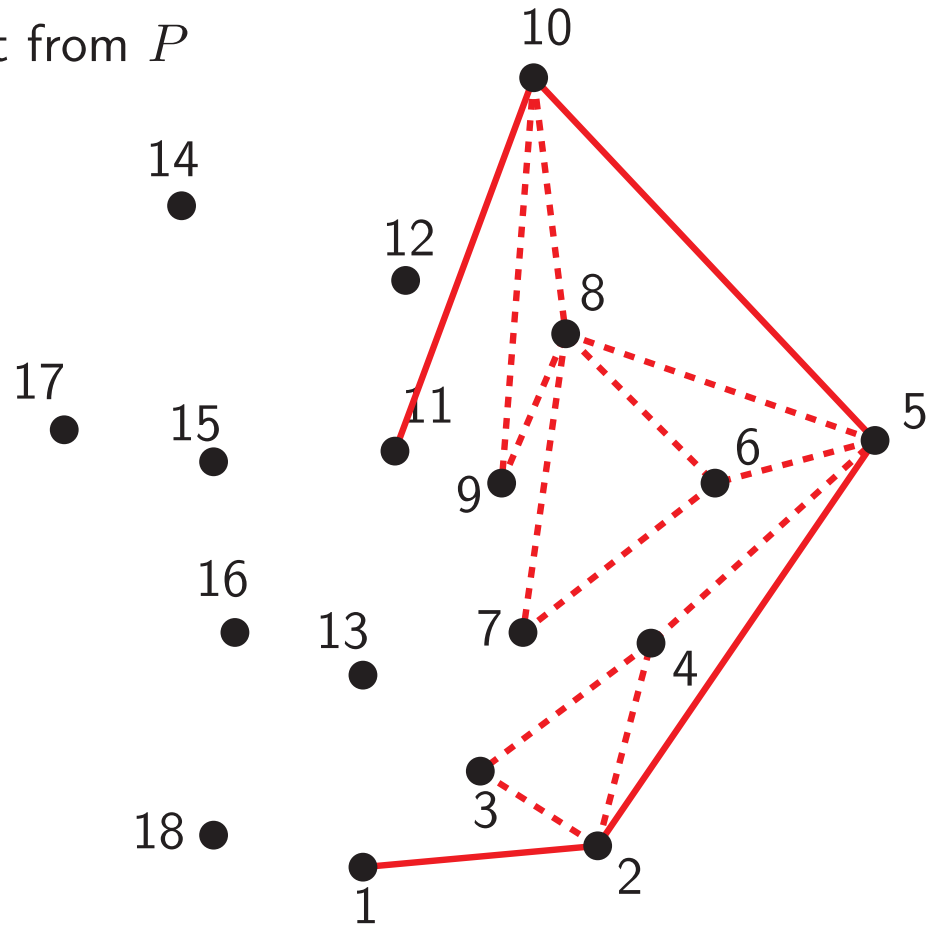
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

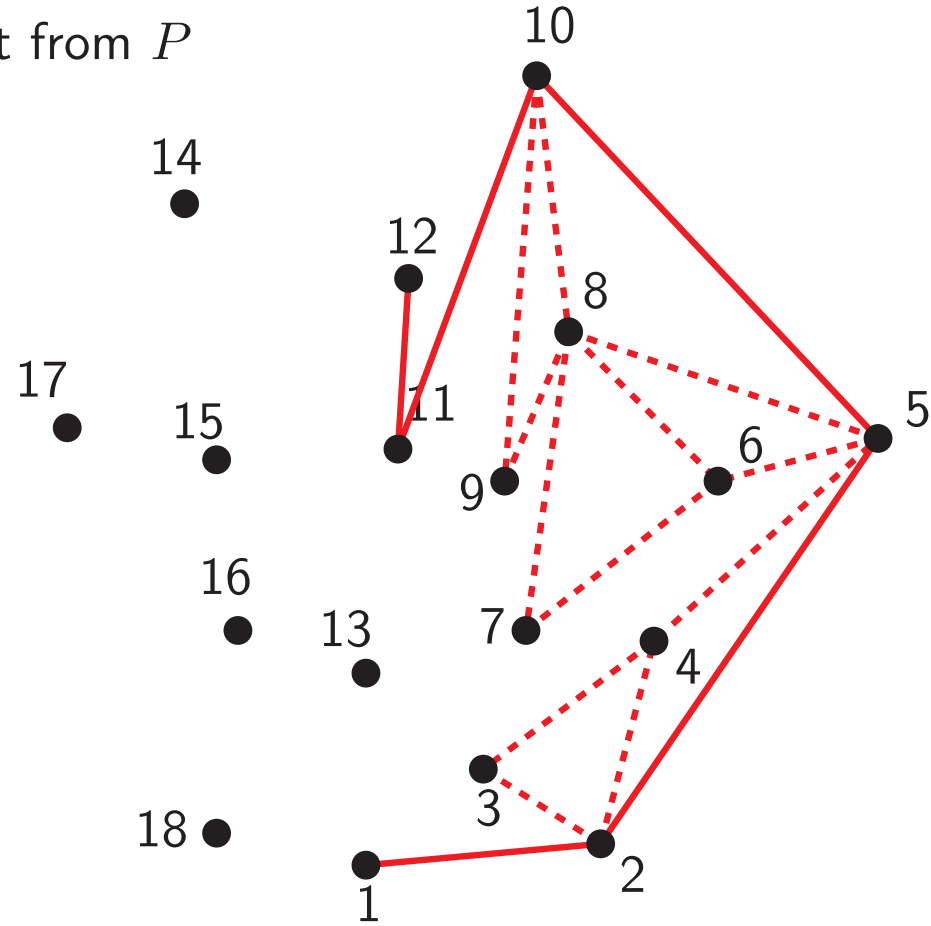
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.



# CONVEX HULL IN 2D

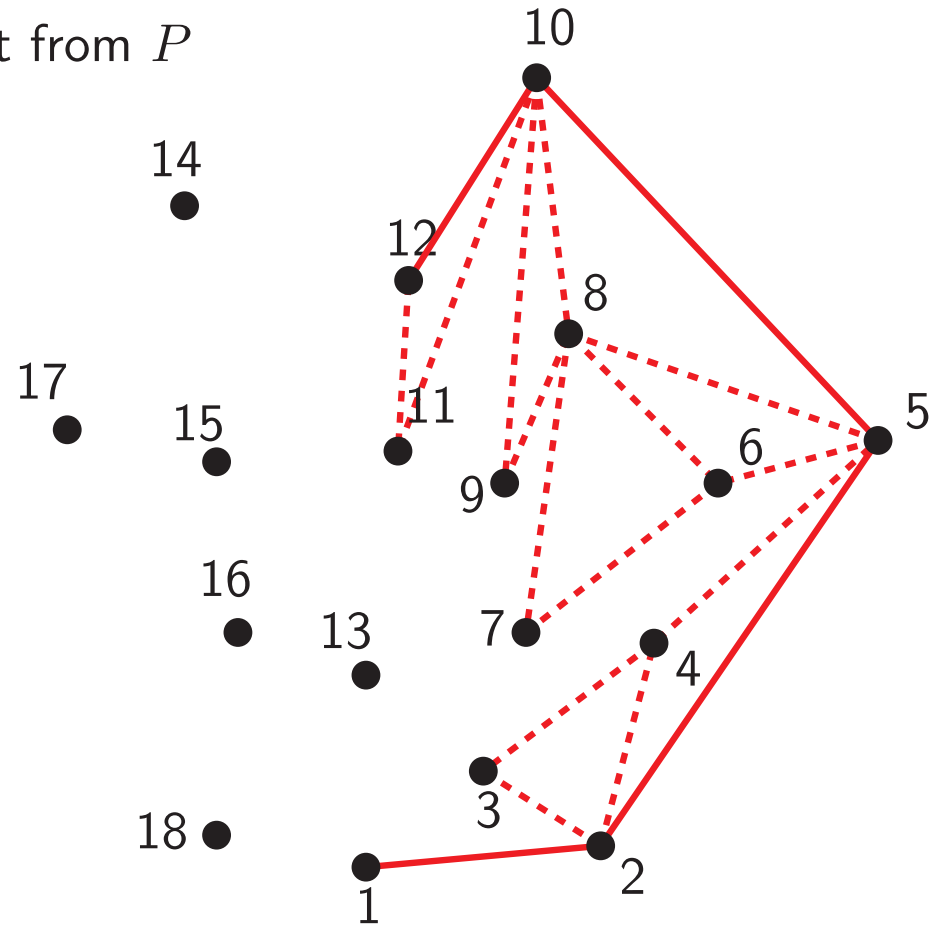
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

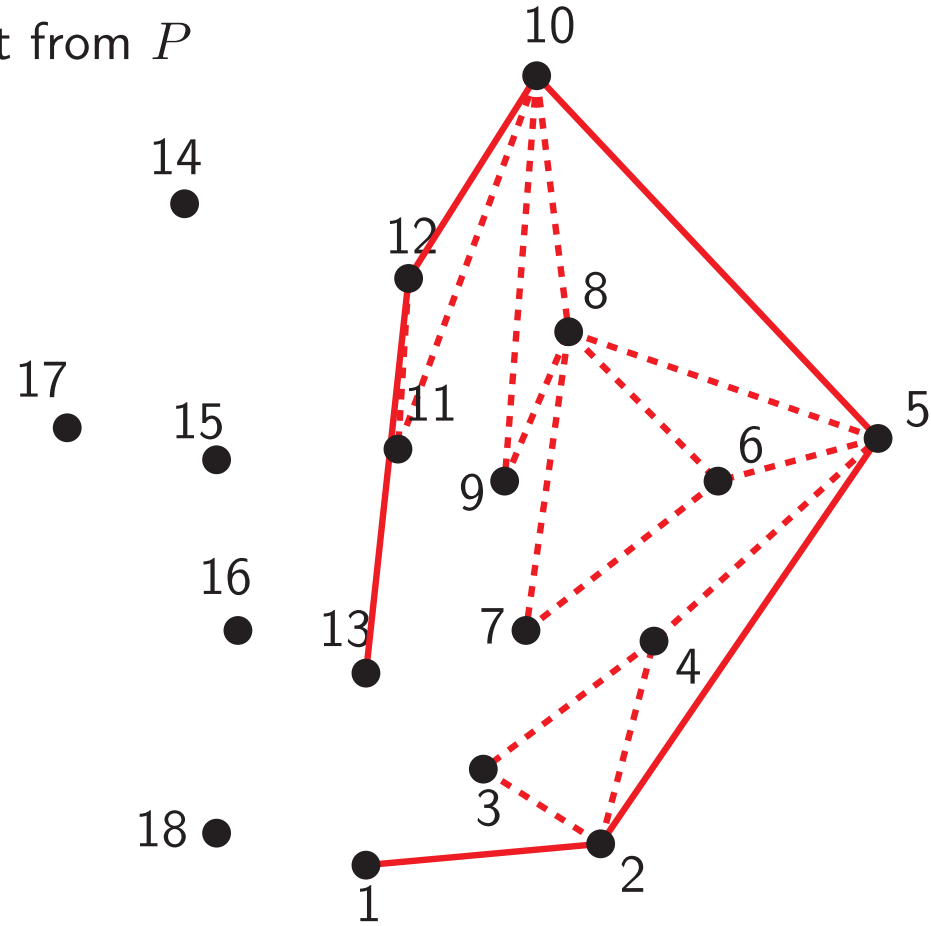
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $l$ 's top point, push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $l$ 's top point, delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

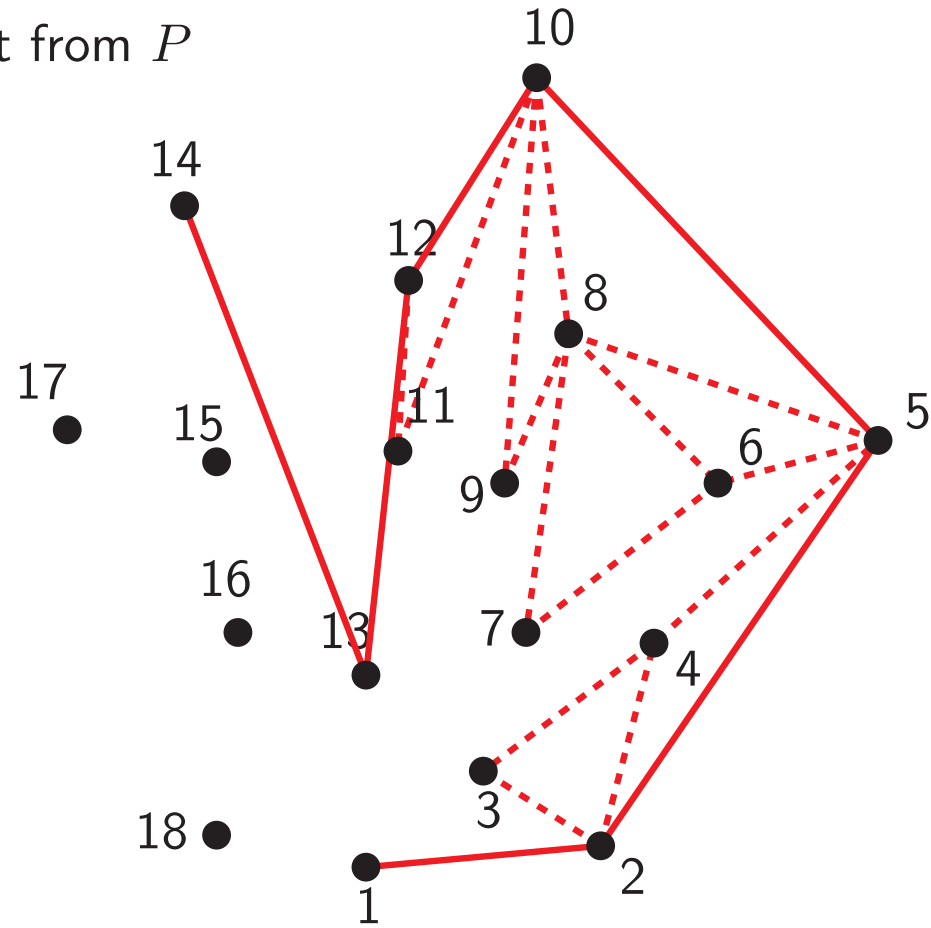
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $l$ 's top element, push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $l$ 's top element, delete  $l$ 's top element and repeat



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

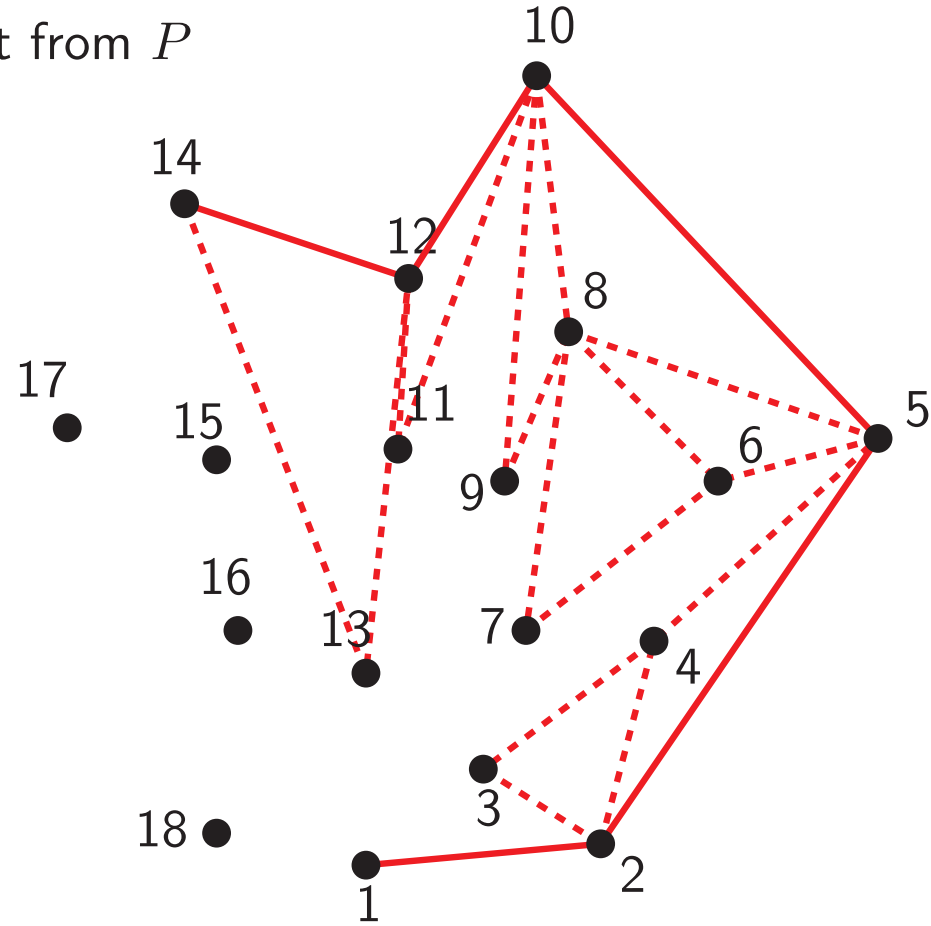
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

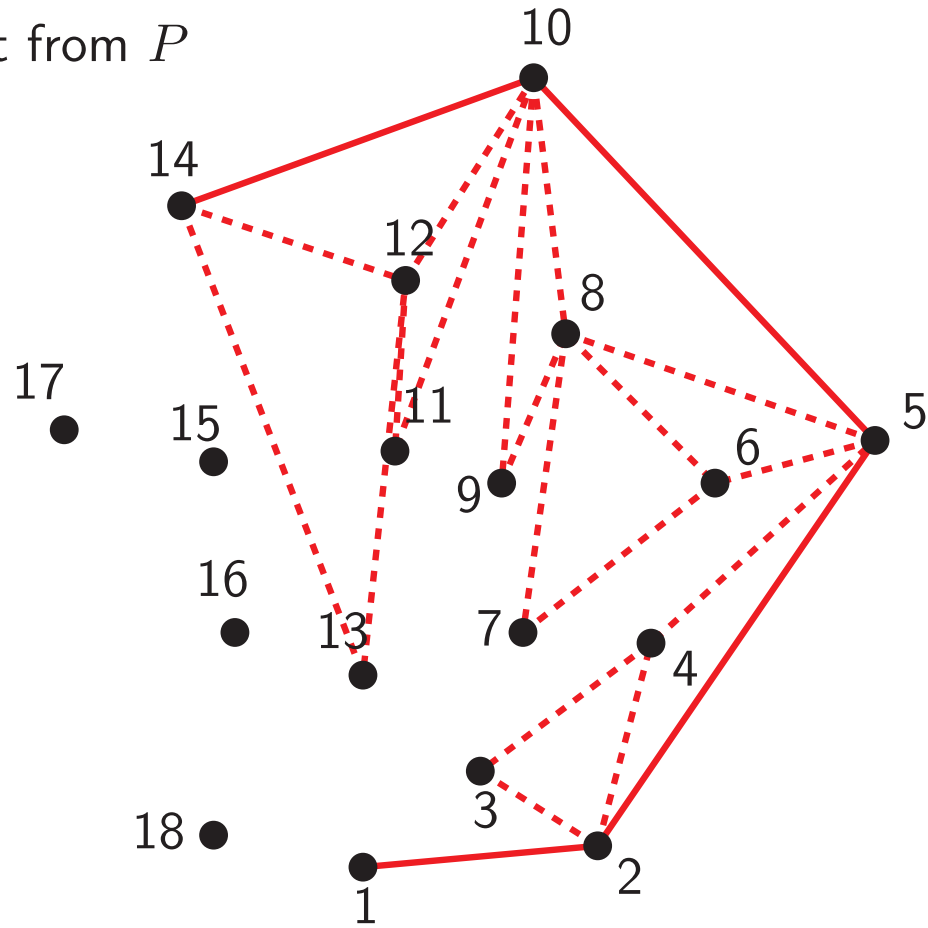
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

At each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is a left turn:
  - Push  $p_i$  in  $l$
  - Delete  $p_{i-1}$  from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
 ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
 CALL OR WHATSAPP: 689 45 44 70

www.cartagenap99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

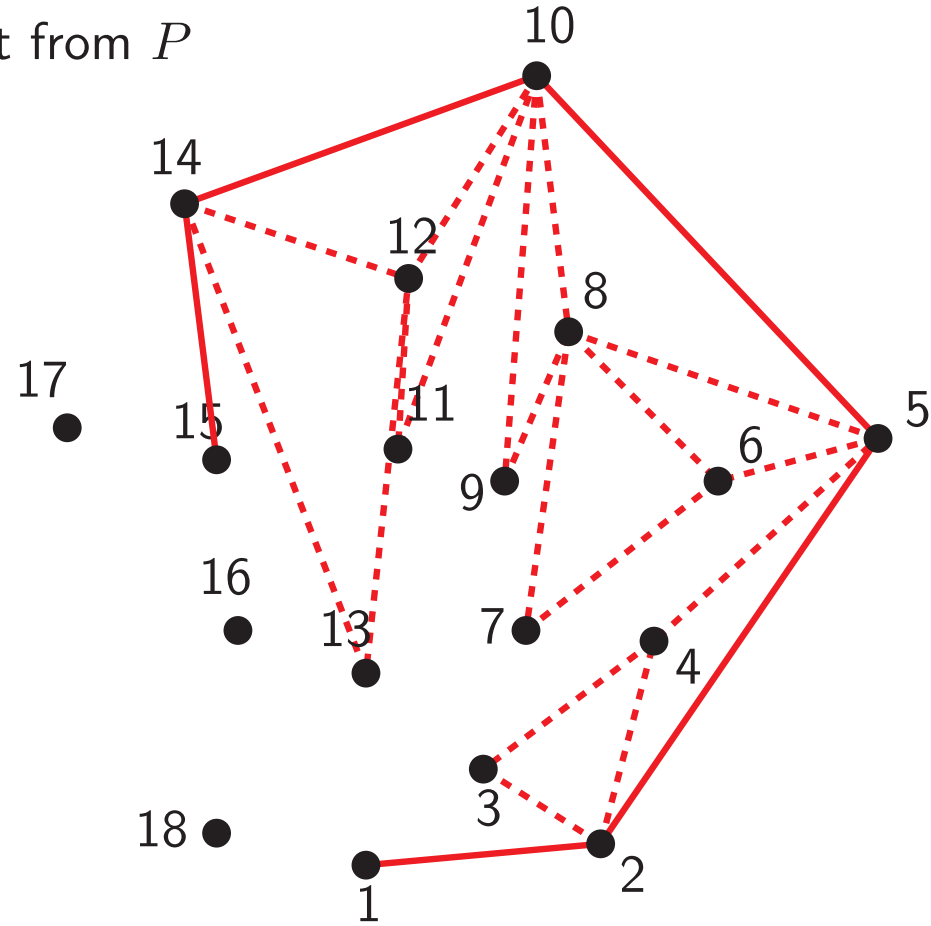
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , push it in  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , delete it from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

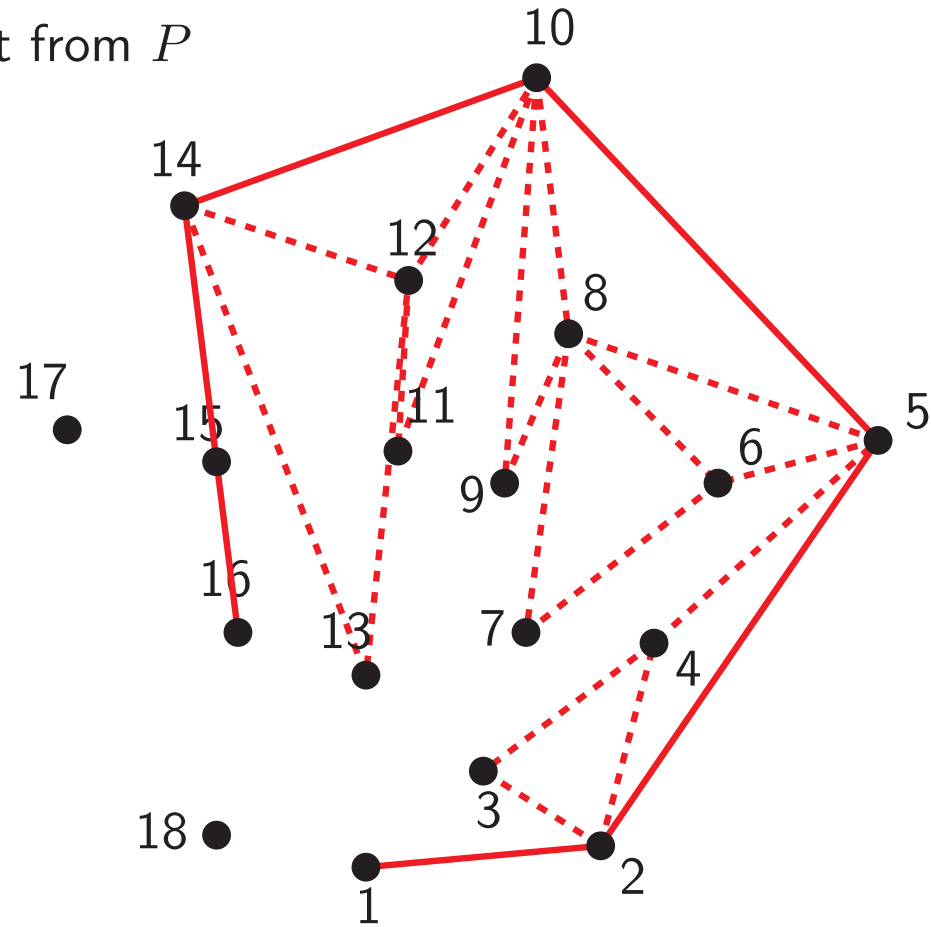
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line formed by  $p_{i-1}$  and  $p_{i-2}$ , push it in  $l$
- Else, pop from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

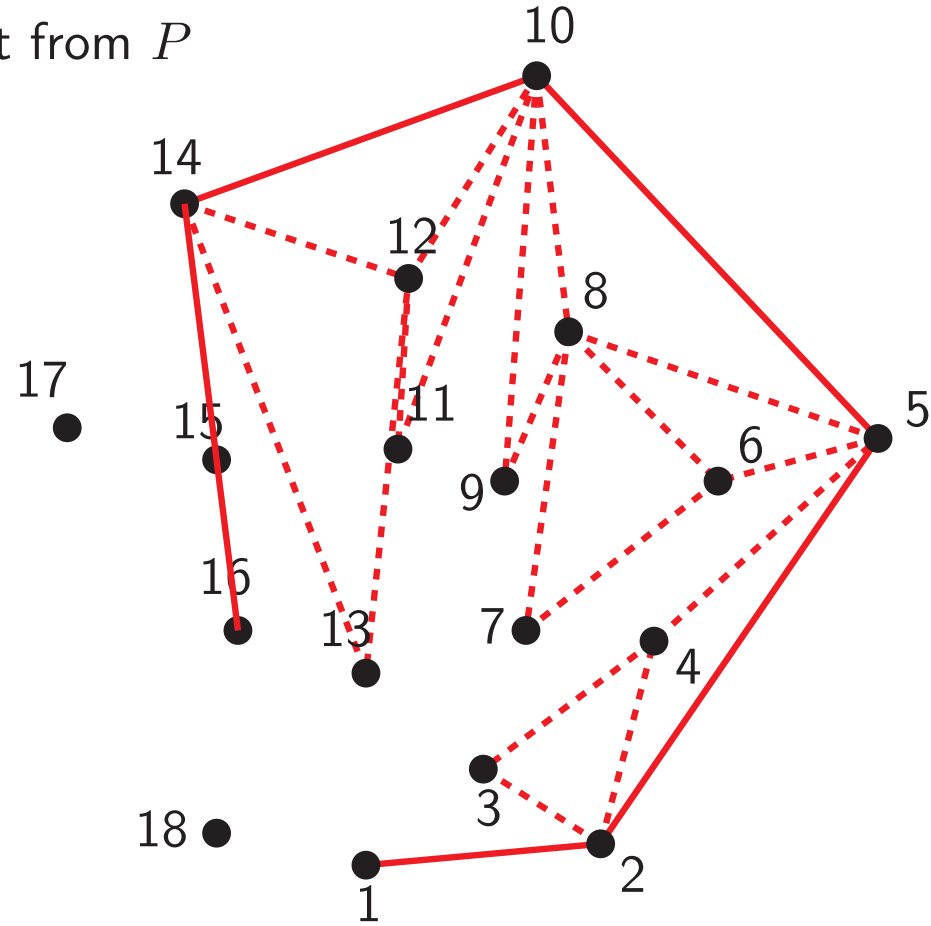
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $\angle(p_i, \text{top}(l))$  is a left turn:
  - Push  $p_i$  in  $l$
  - Delete  $p_i$  from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.



# CONVEX HULL IN 2D

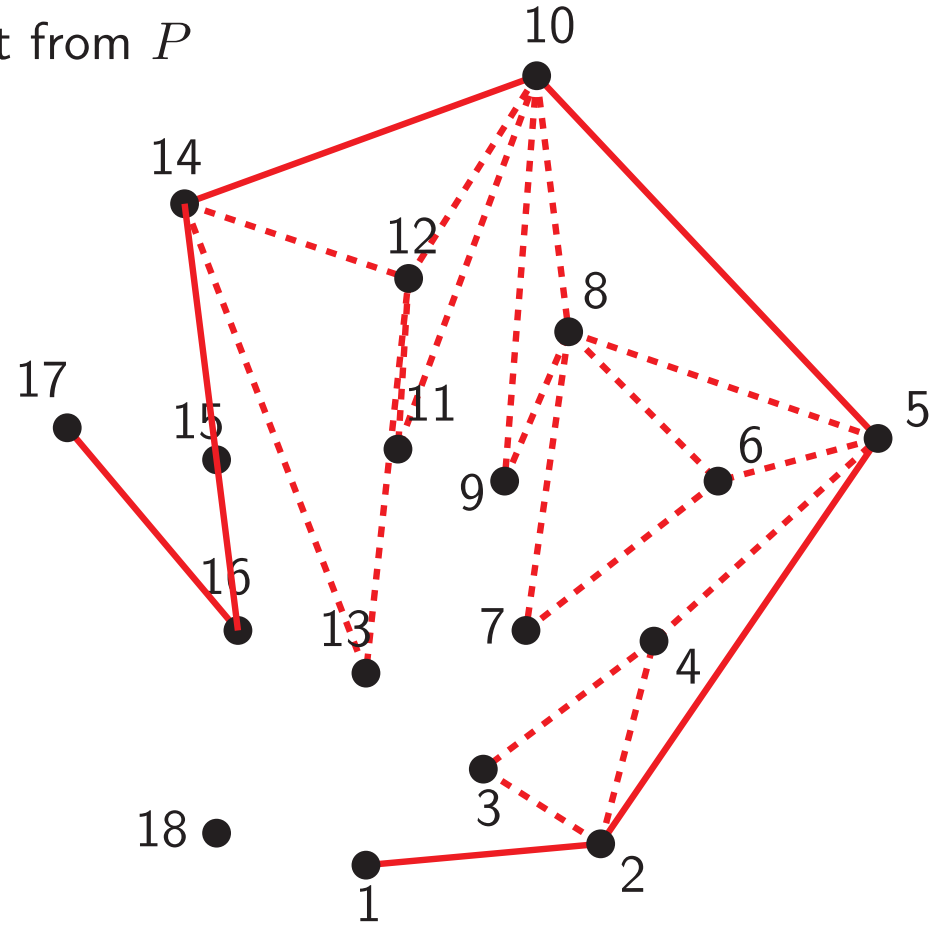
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

- If  $\angle(p_i, \text{top}(l))$  is a left turn:
  - Push  $p_i$  in  $l$
  - Delete  $p_i$  from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

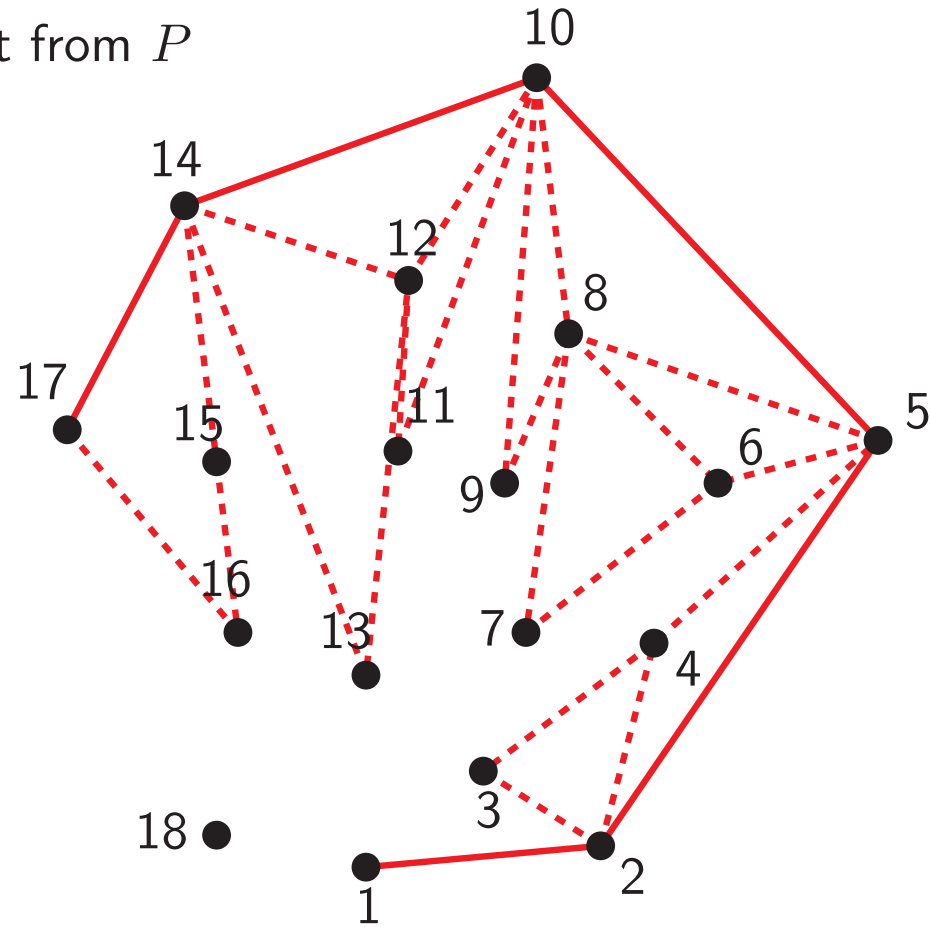
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

At each point  $p_i \in P$  to be explored, do:

- If  $\angle(p_i, \text{top}(l))$  is a left turn:
  - Push  $p_i$  in  $l$
  - Delete  $p_i$  from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

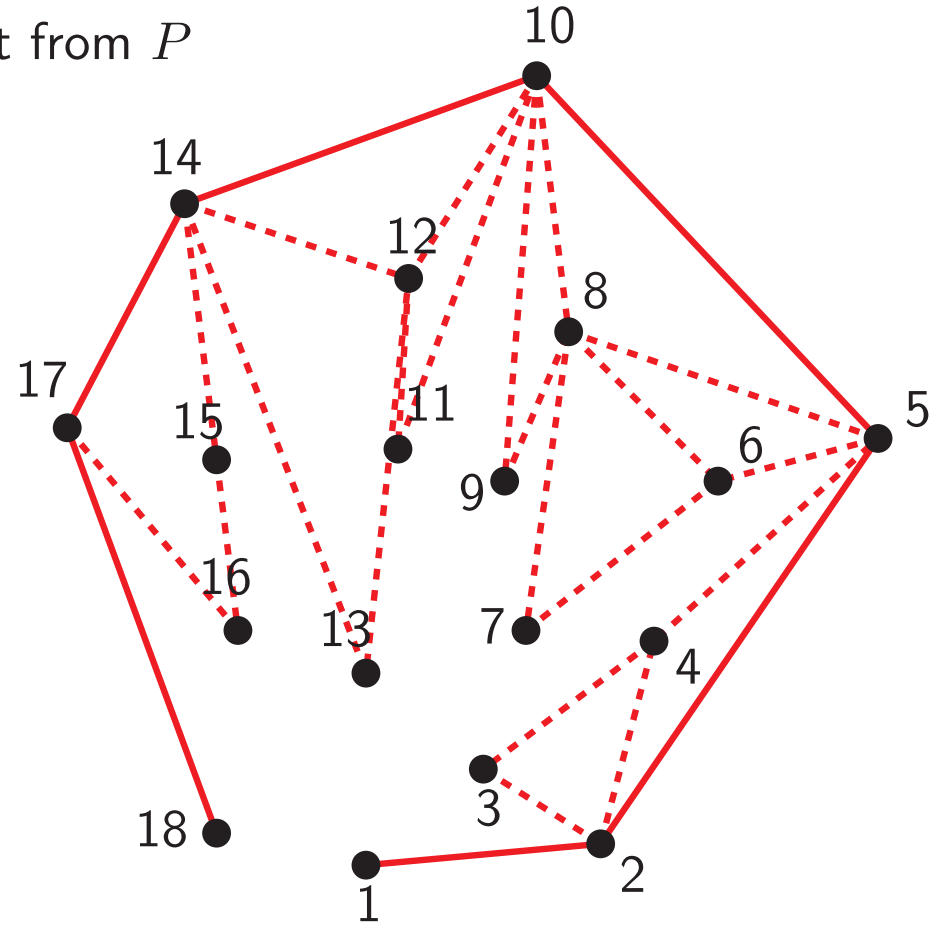
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

At each point  $p_i \in P$  to be explored, do:

- If  $\angle(p_i, \text{top}(l))$  is a left turn:
  - Push  $p_i$  in  $l$
  - Delete  $p_i$  from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Add point in  $l$  and delete if from  $P$

At points  $p_i \in P$  to be explored, do:

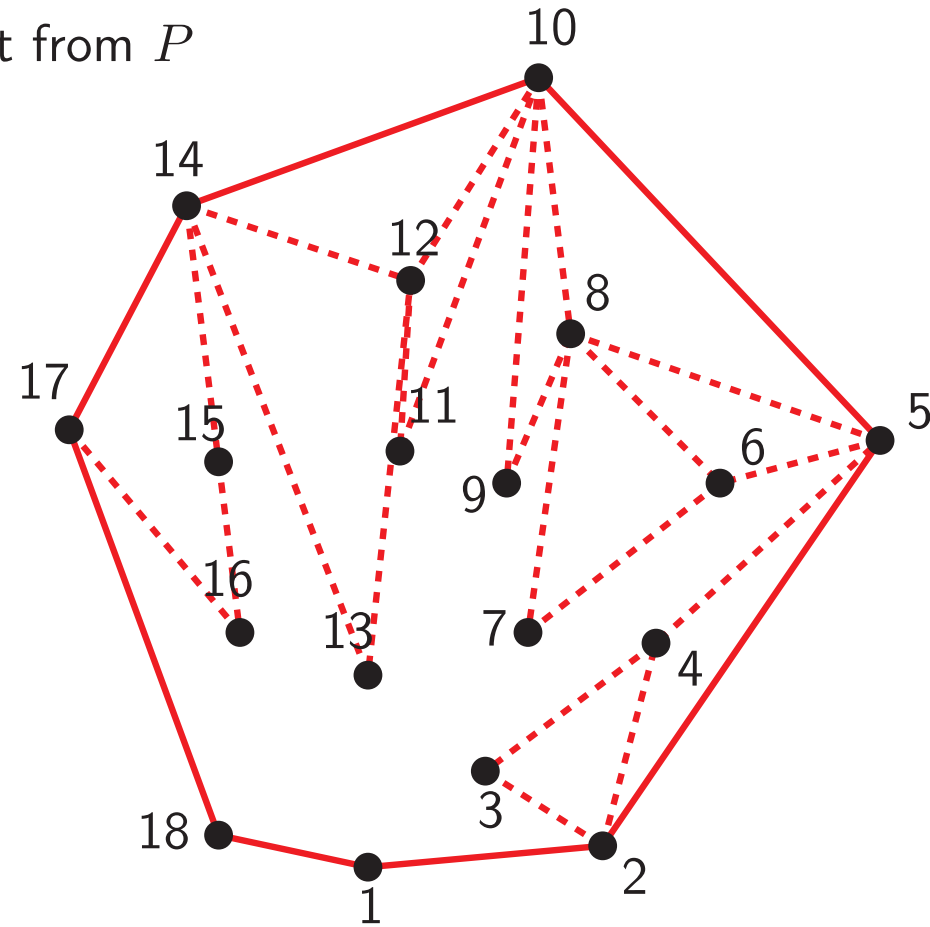
if  $\angle(p_{i-1}, p_i, p_{i+1}) < \pi$

    make left turn:

    push  $p_i$  in  $l$

    delete  $p_i$  from  $P$

return  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

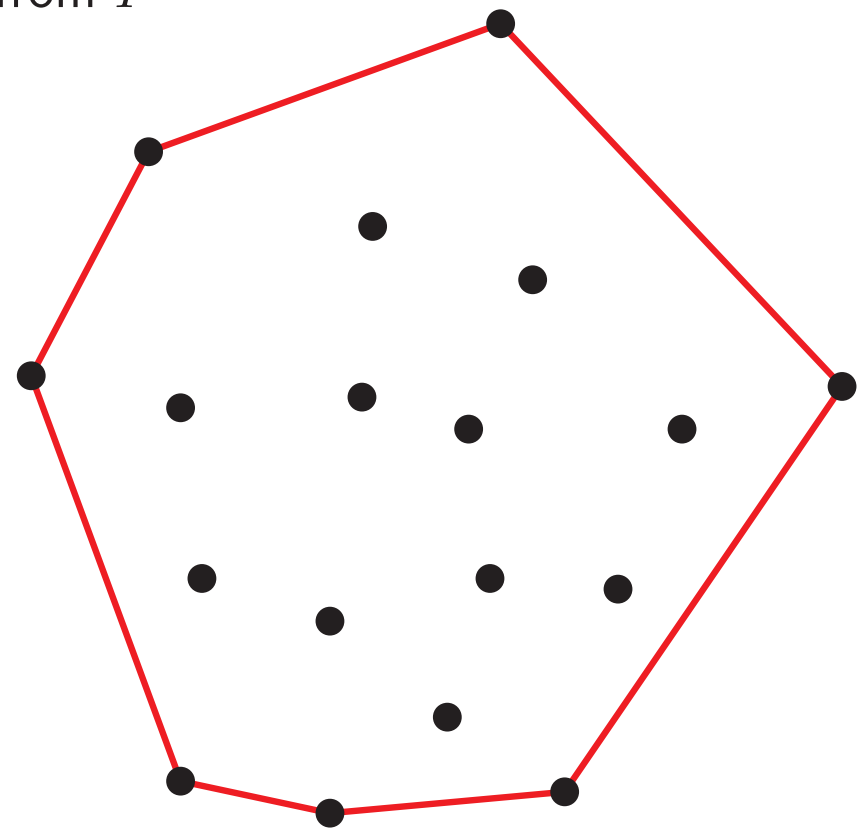
## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- Add the next point in  $l$  and delete it from  $P$

At each point  $p_i \in P$  to be explored, do:

- If  $p_i$  is to the left of the line through  $l$  and  $p_{i-1}$ , then push  $p_i$  on  $l$
- Else if  $p_i$  is on the line through  $l$  and  $p_{i-1}$ , then delete  $p_i$  from  $l$
- Else if  $p_i$  is to the right of the line through  $l$  and  $p_{i-1}$ , then delete  $p_i$  from  $l$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
 ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
 CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Graham's algorithm

### Initialization

- Find a vertex  $v$  of  $ch(P)$ , push it in  $l$  and delete it from  $P$
- Angularly sort the points around  $v$
- For each point in  $l$  and delete if from  $P$

For each point  $p_i \in P$  to be explored, do:

...  $s(\text{top}(l))$

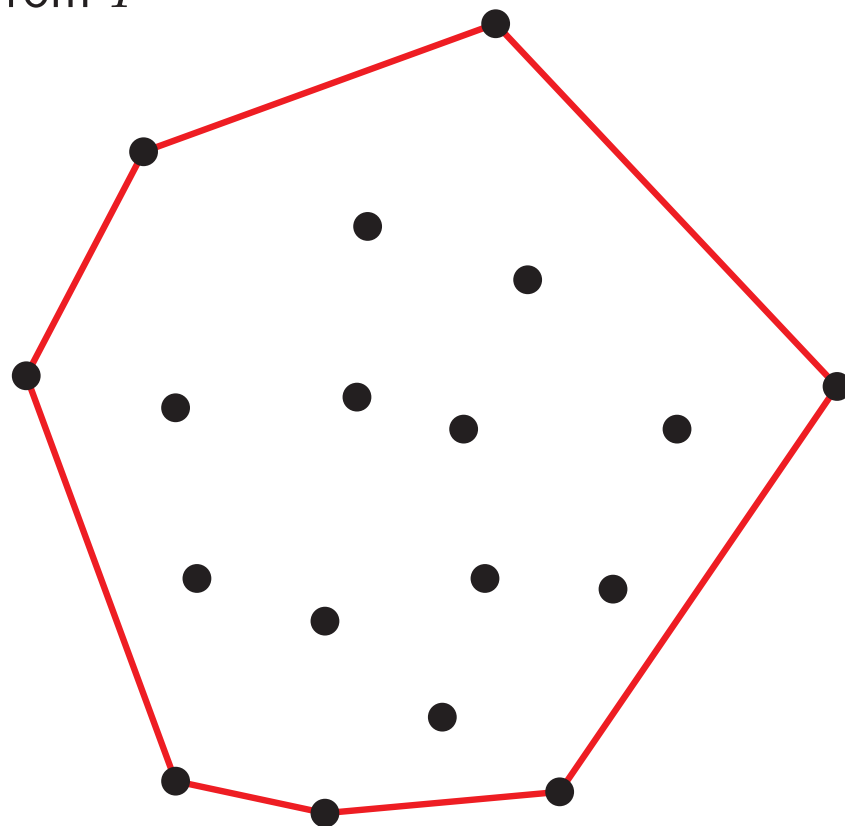
... left turn:

... in  $l$

...  $e_i$

... from  $l$

...  $(n \log n)$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagenas99.com no se hace responsable de la información contenida en el presente documento en virtud al artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

--

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

the points  $p_l$  and  $p_r$

the supporting lines

$p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

Adv  
Fr

do:

the exterior of the polygon defined by  $l$ :

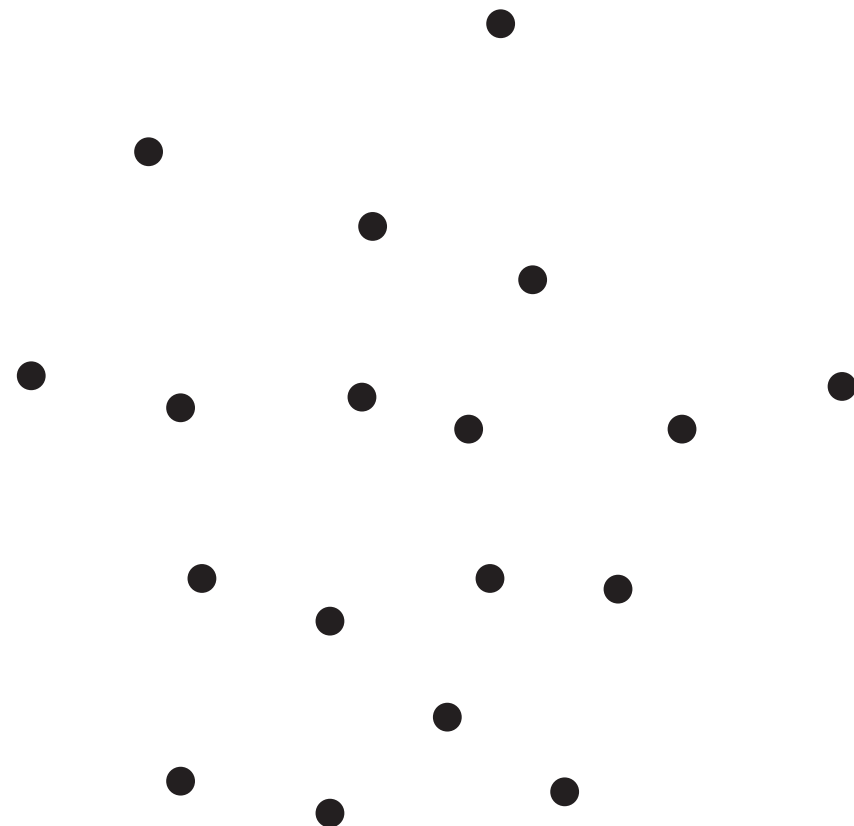
the points  $p_l$  and  $p_r$

the supporting lines

$p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# CONVEX HULL IN 2D

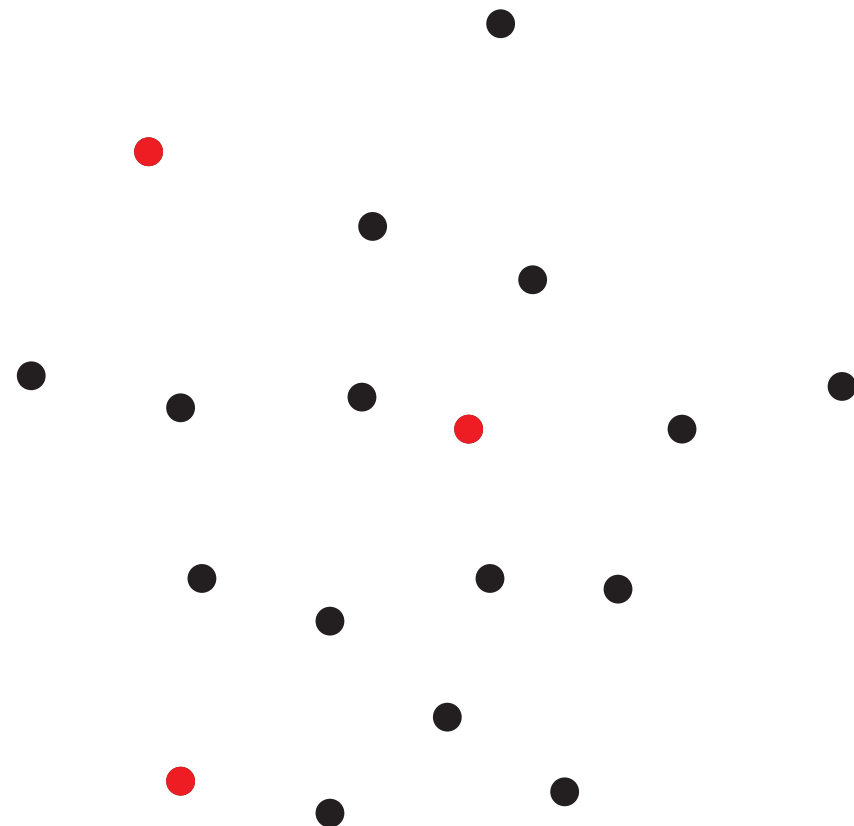
## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

Adv  
Fr

do:  
the exterior of the polygon defined by  $l$ :  
the points  $p_l$  and  $p_r$   
the supporting lines  
the chain  $p_l, \dots, p_r$  in  $l$   
the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

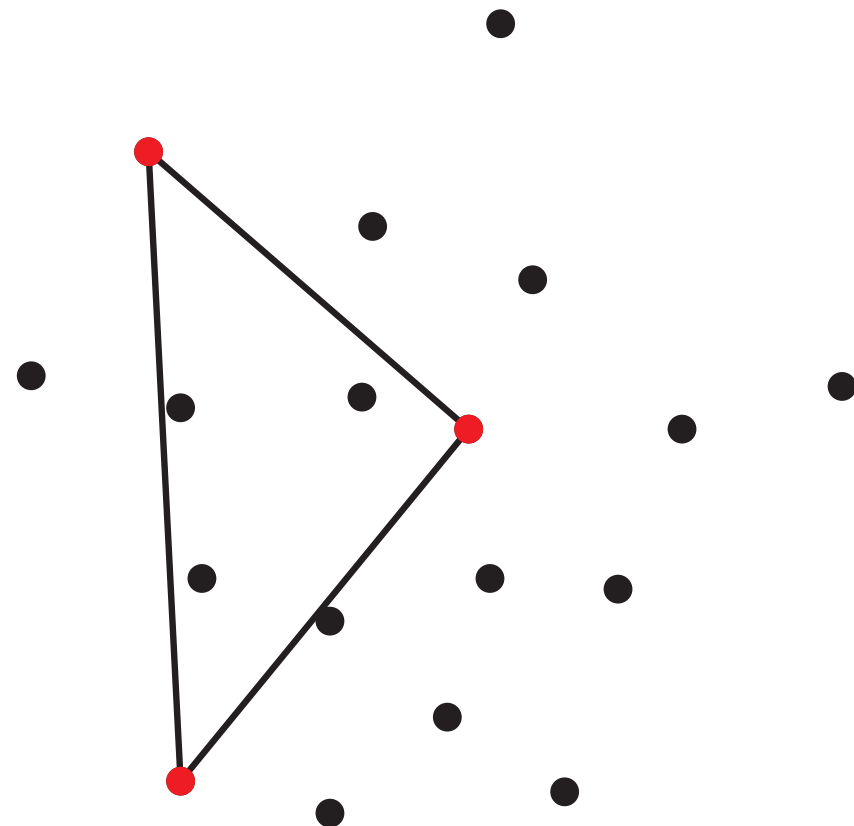
## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

Adv  
Fr

do:  
the exterior of the polygon defined by  $l$ :  
the points  $p_l$  and  $p_r$   
the supporting lines  
the chain  $p_l, \dots, p_r$  in  $l$   
the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

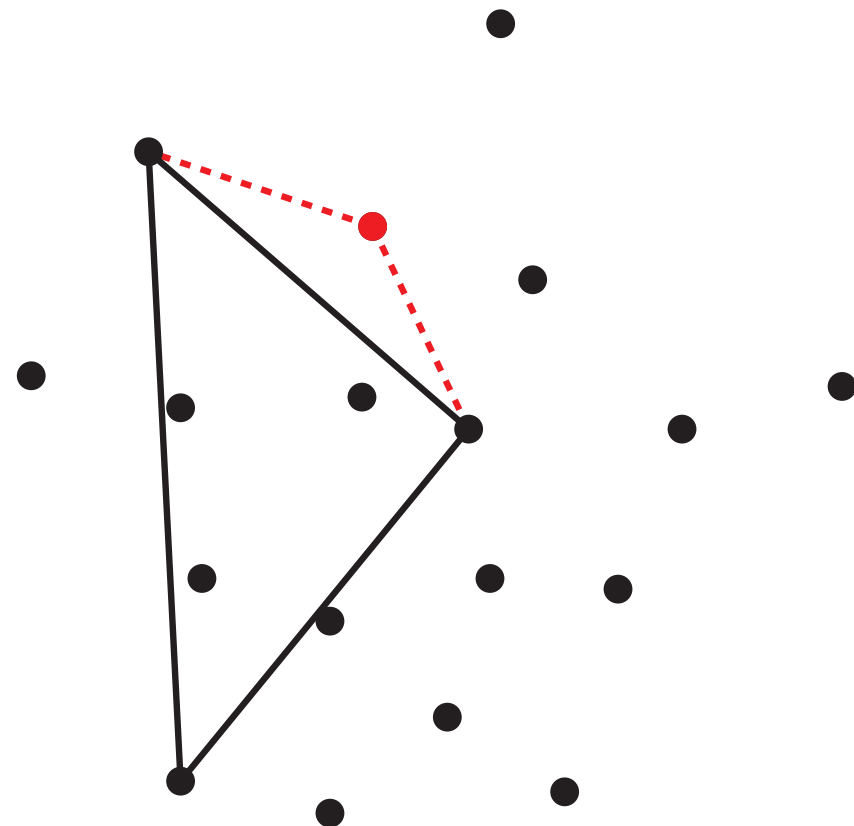
the points  $p_l$  and  $p_r$

the supporting lines

the points  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

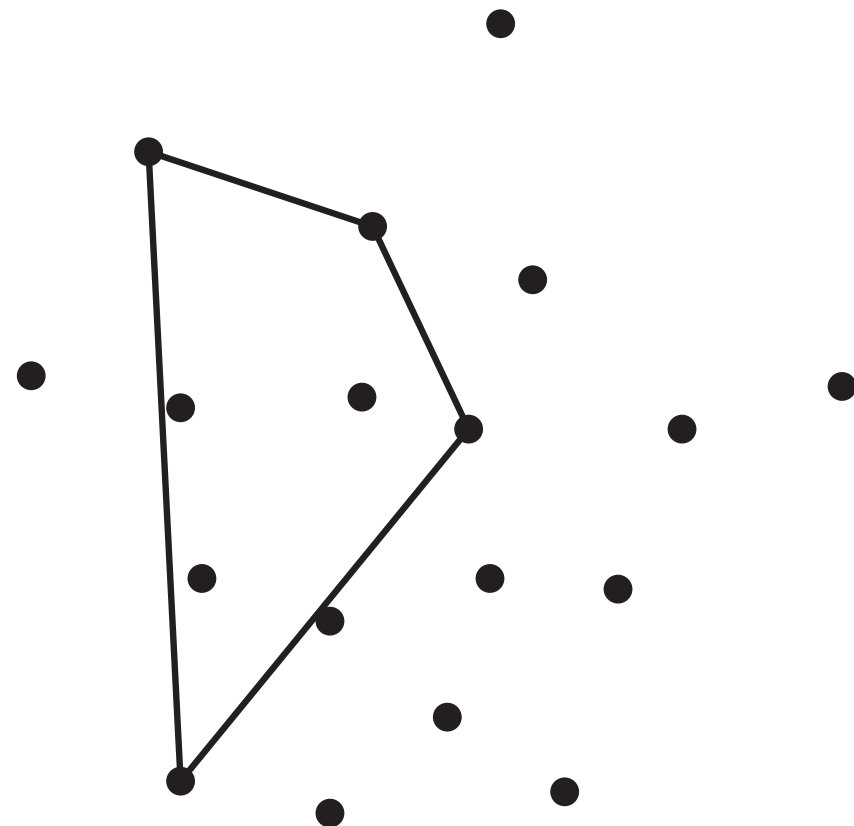
the points  $p_l$  and  $p_r$

the supporting lines

$p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

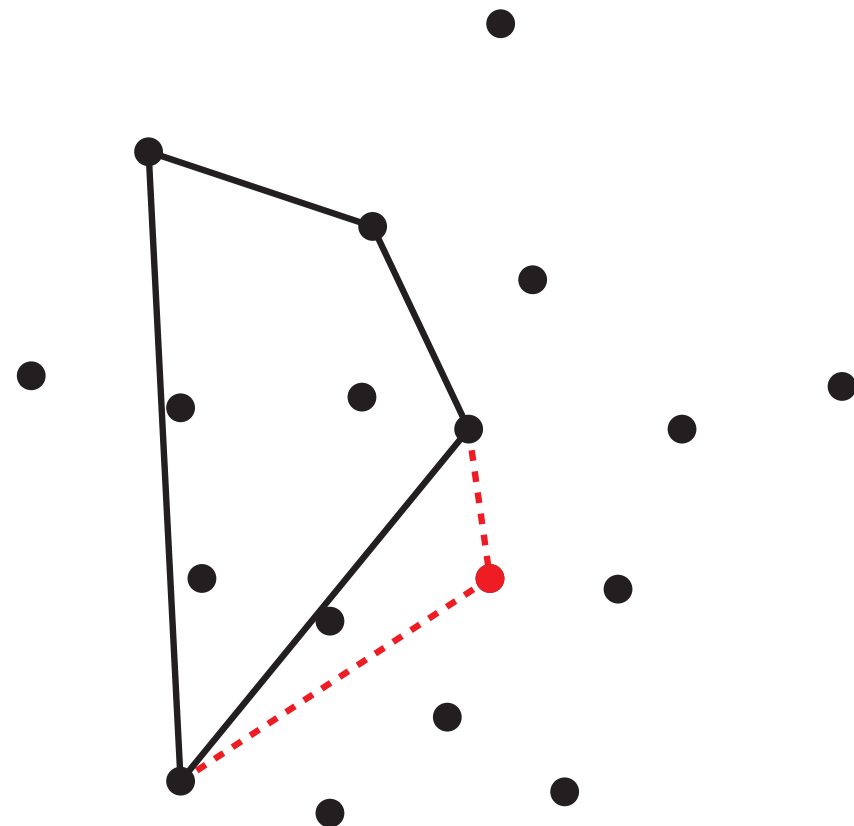
## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

Advances in  
Fractal Geometry

do:  
the exterior of the polygon defined by  $l$ :  
the points  $p_l$  and  $p_r$   
the supporting lines  
the points  $p_i$  to the polygon  
the chain  $p_l, \dots, p_r$  in  $l$   
the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

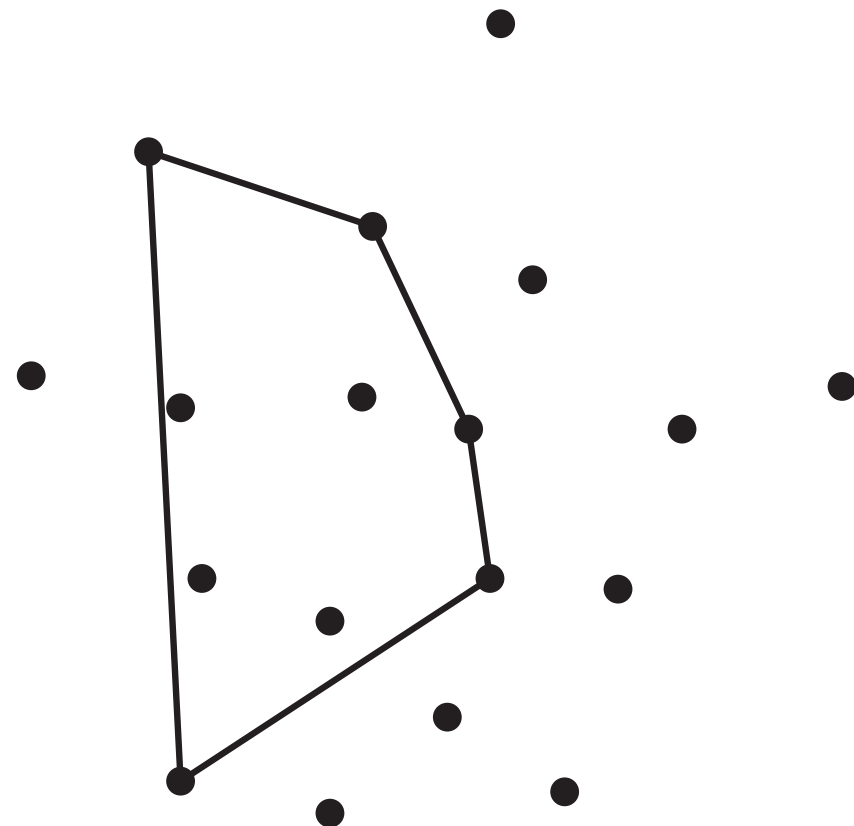
the points  $p_l$  and  $p_r$

the supporting lines

the point  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

Adv: For each point  $p_i$ , do:

if  $p_i$  is in the exterior of the polygon defined by  $l$ :

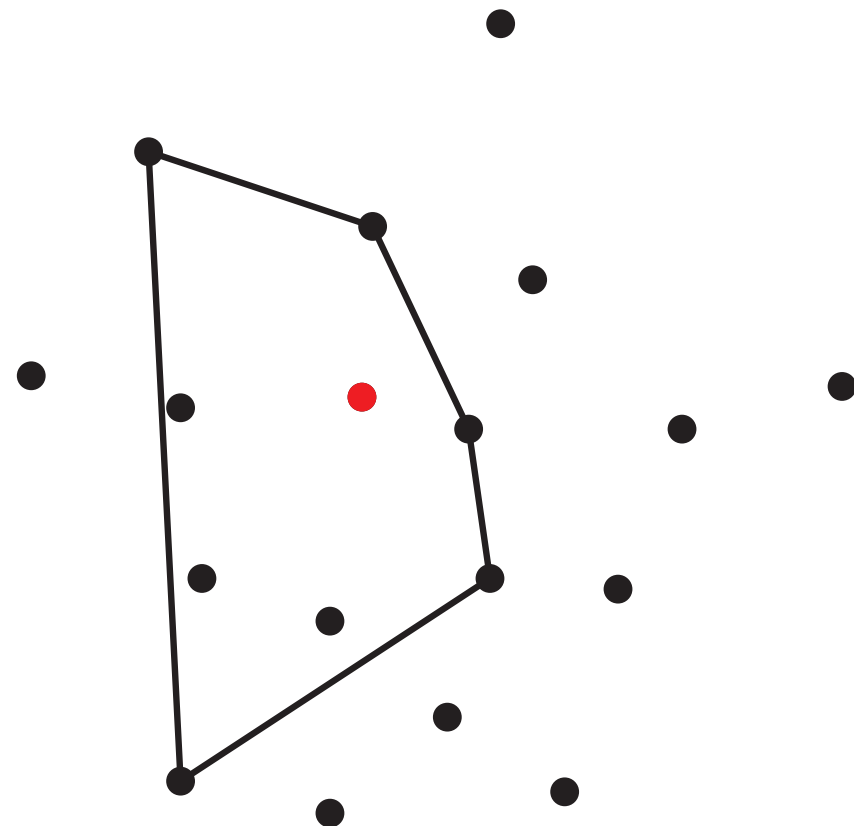
if  $p_i$  is between the points  $p_l$  and  $p_r$

if  $p_i$  is between the supporting lines

if  $p_i$  is to the right of the line  $p_l p_i$

if  $p_i$  is to the left of the line  $p_i p_r$

if  $p_i$  is to the left of the line  $p_l p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

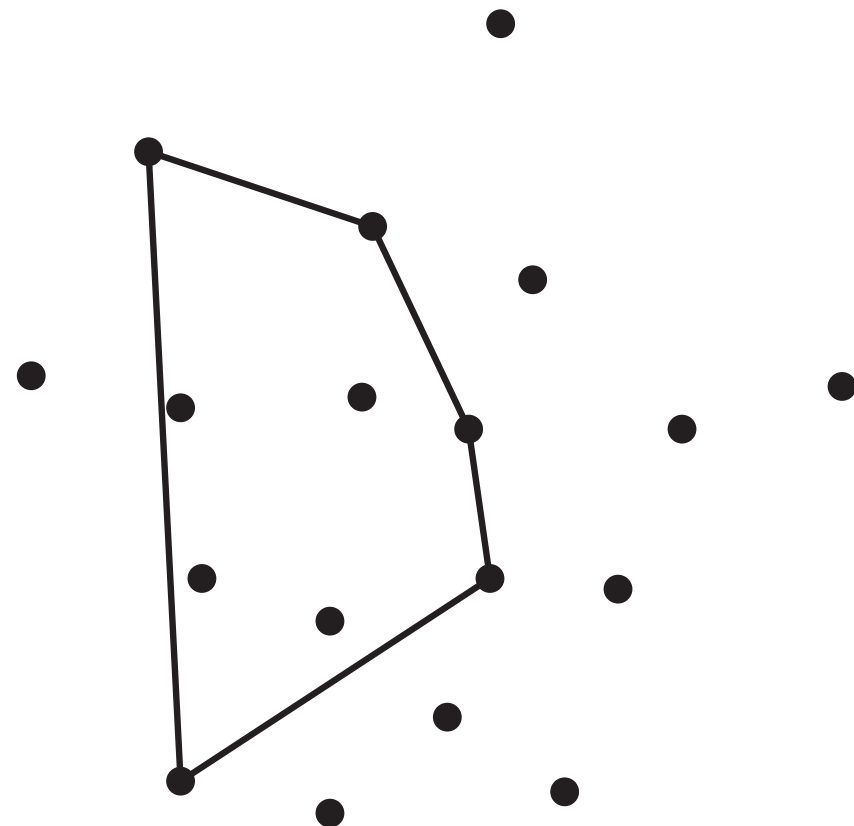
the points  $p_l$  and  $p_r$

the supporting lines

the point  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

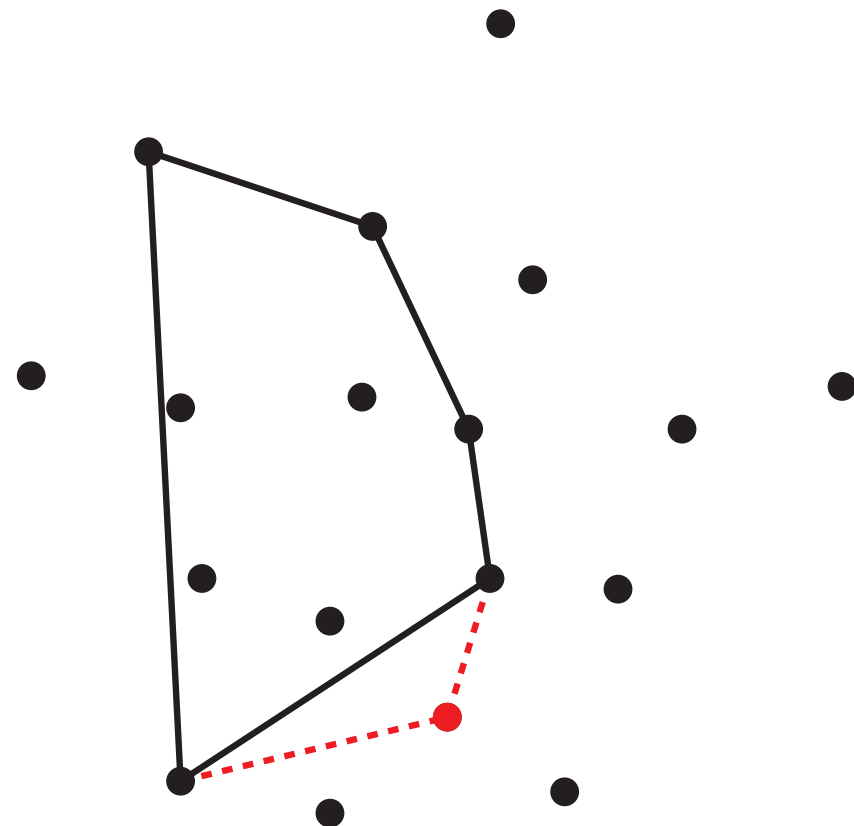
the exterior of the polygon defined by  $l$ :

the points  $p_l$  and  $p_r$

the supporting lines

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

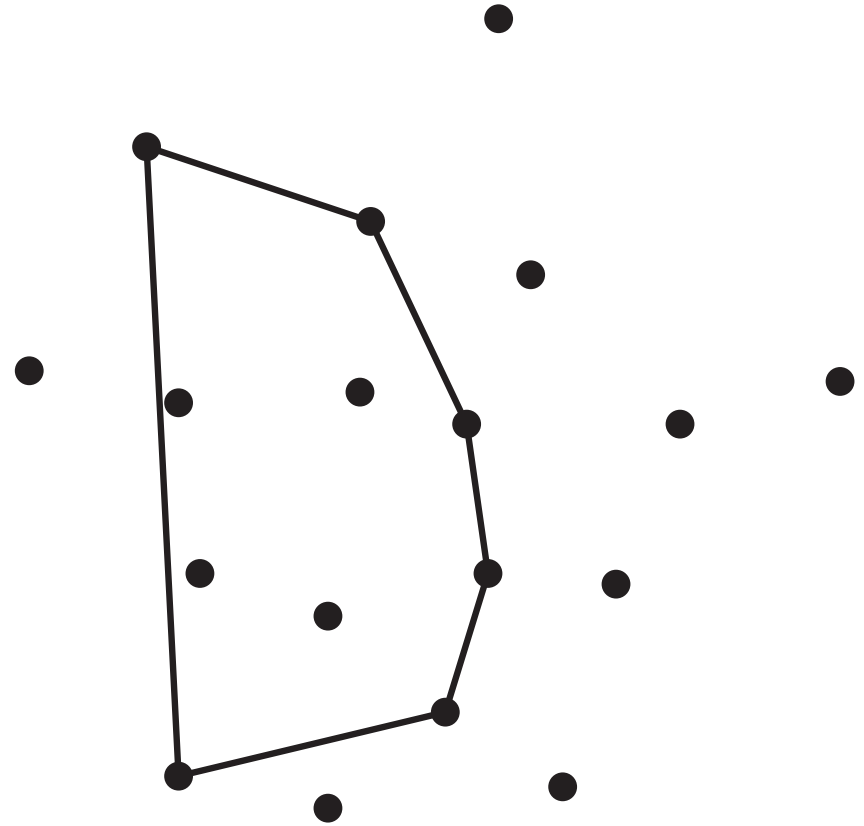
the points  $p_l$  and  $p_r$

the supporting lines

the point  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

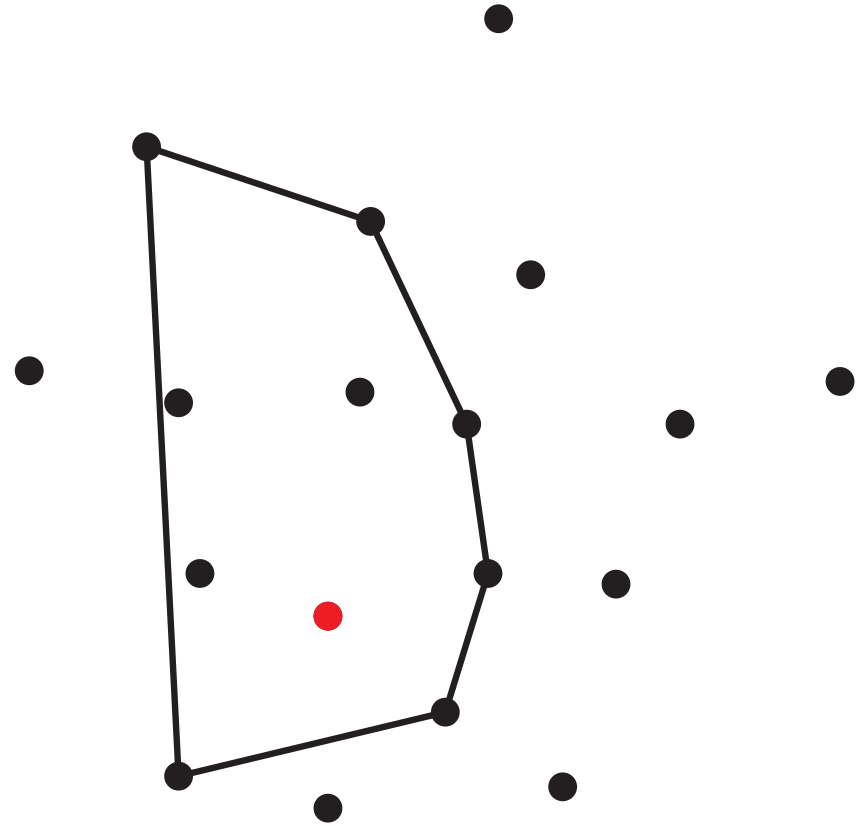
the exterior of the polygon defined by  $l$ :

the points  $p_l$  and  $p_r$

the supporting lines

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

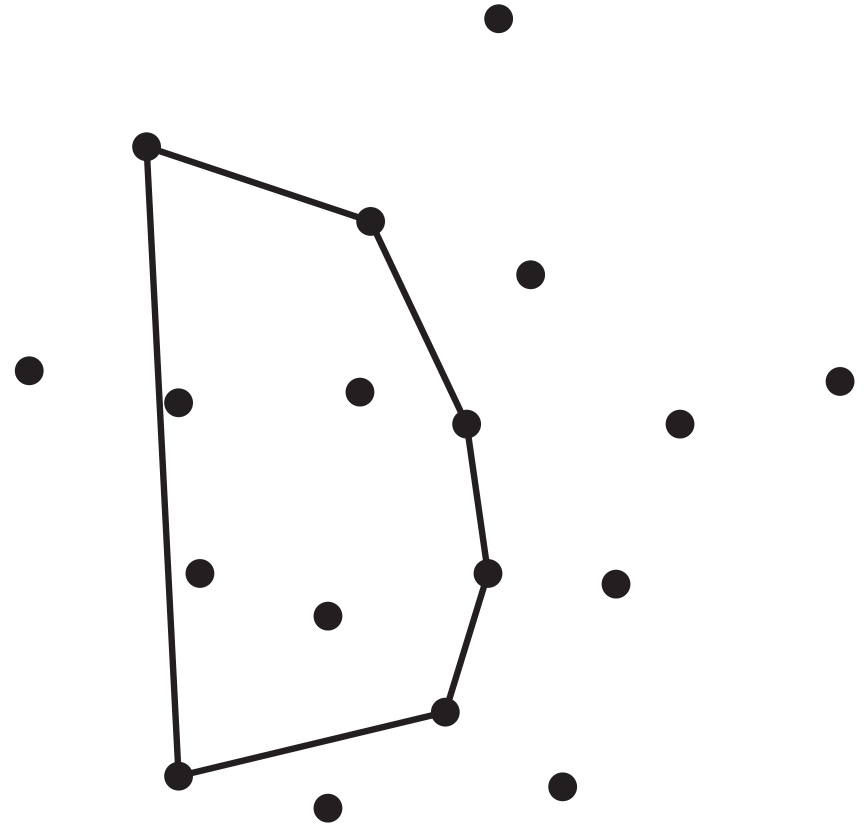
the points  $p_l$  and  $p_r$

the supporting lines

the point  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

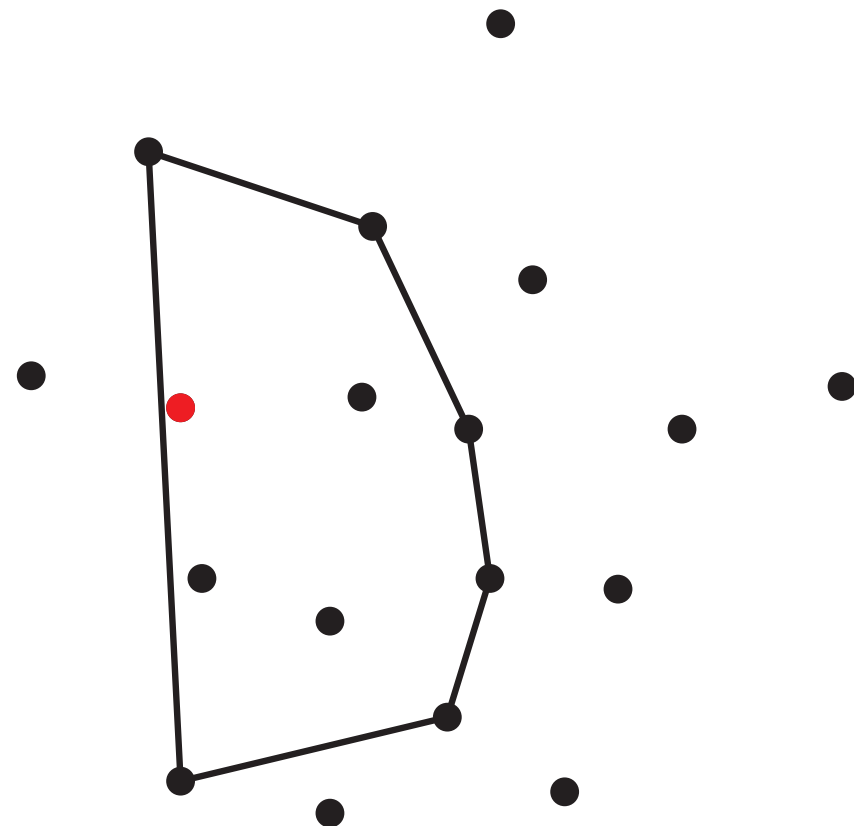
the exterior of the polygon defined by  $l$ :

the points  $p_l$  and  $p_r$

the supporting lines

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

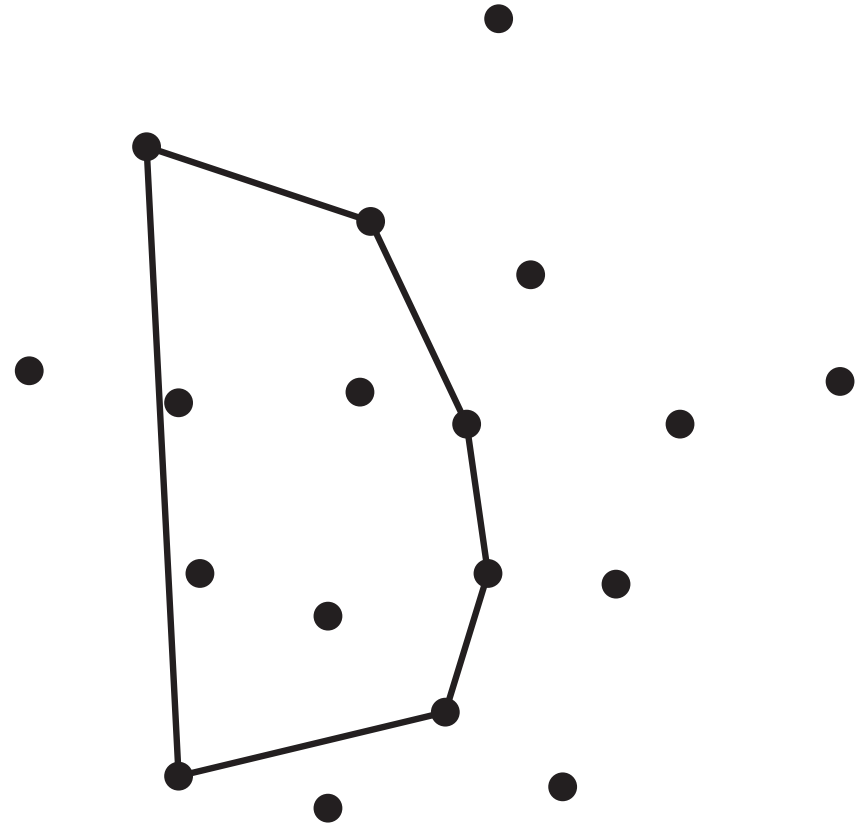
the points  $p_l$  and  $p_r$

the supporting lines

the points  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

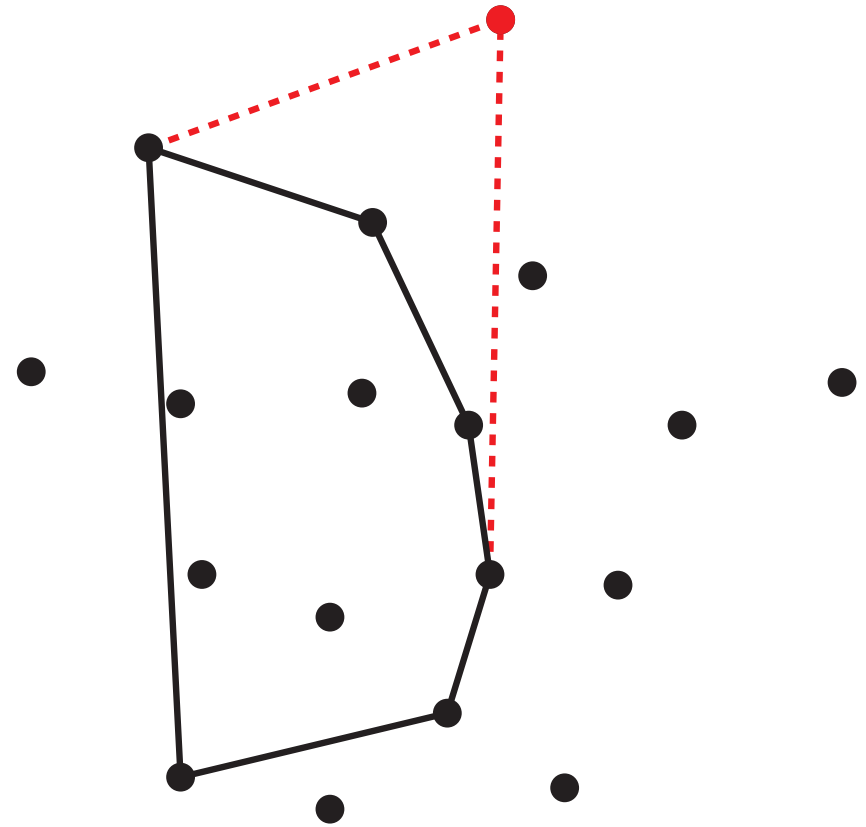
the points  $p_l$  and  $p_r$

the supporting lines

the points  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

Adv  
Fr

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

et

do:

the exterior of the polygon defined by  $l$ :

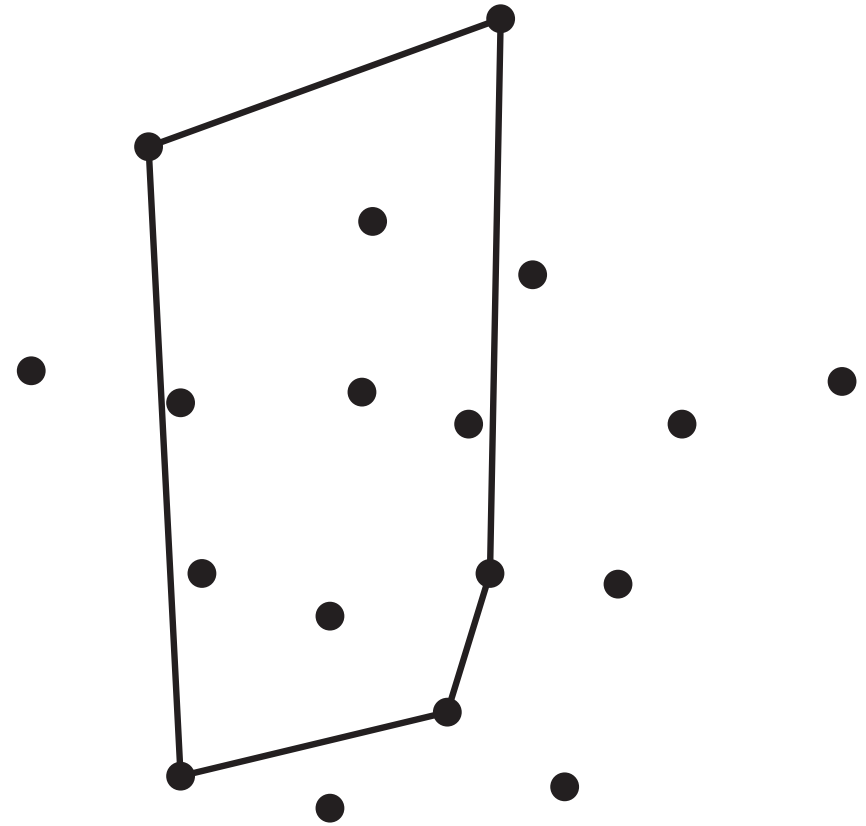
the points  $p_l$  and  $p_r$

the supporting lines

the points  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

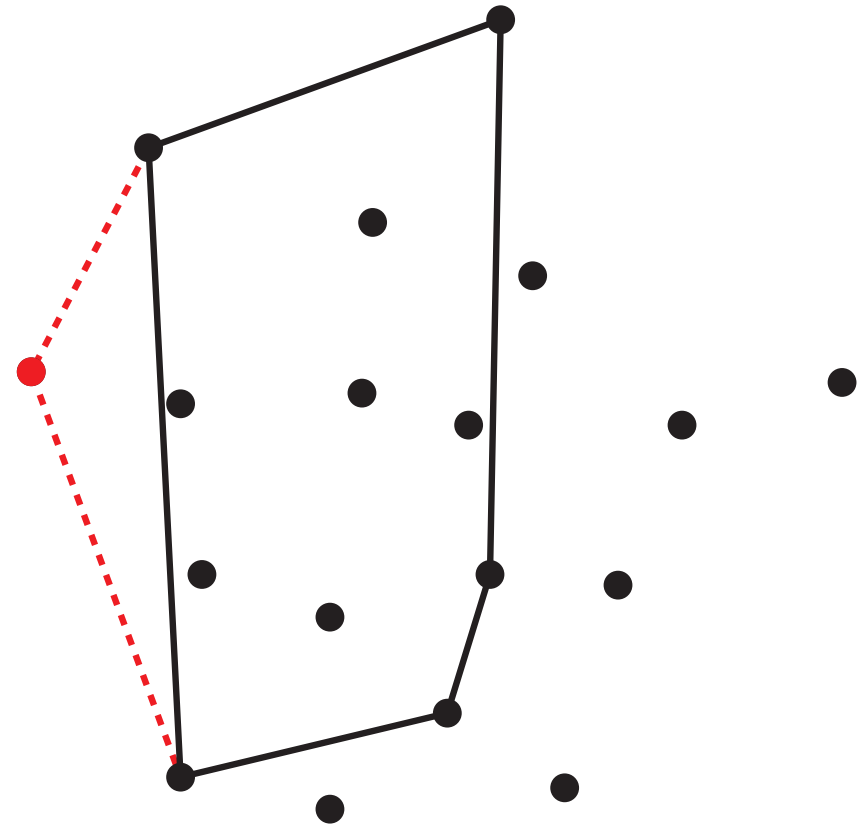
the points  $p_l$  and  $p_r$

the supporting lines

$p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

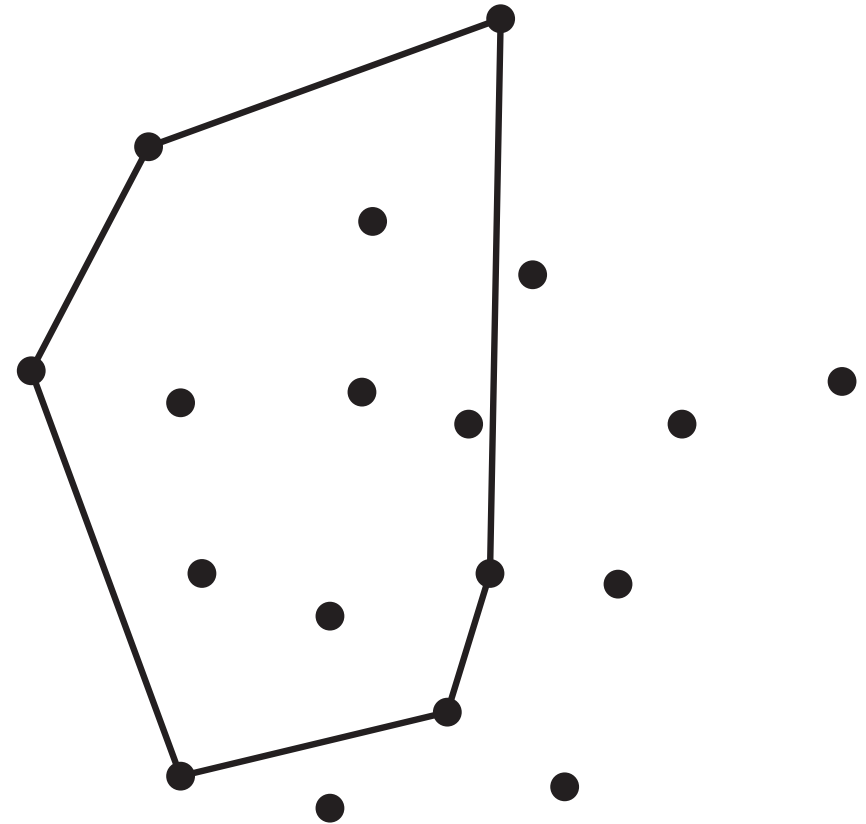
the points  $p_l$  and  $p_r$

the supporting lines

$p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

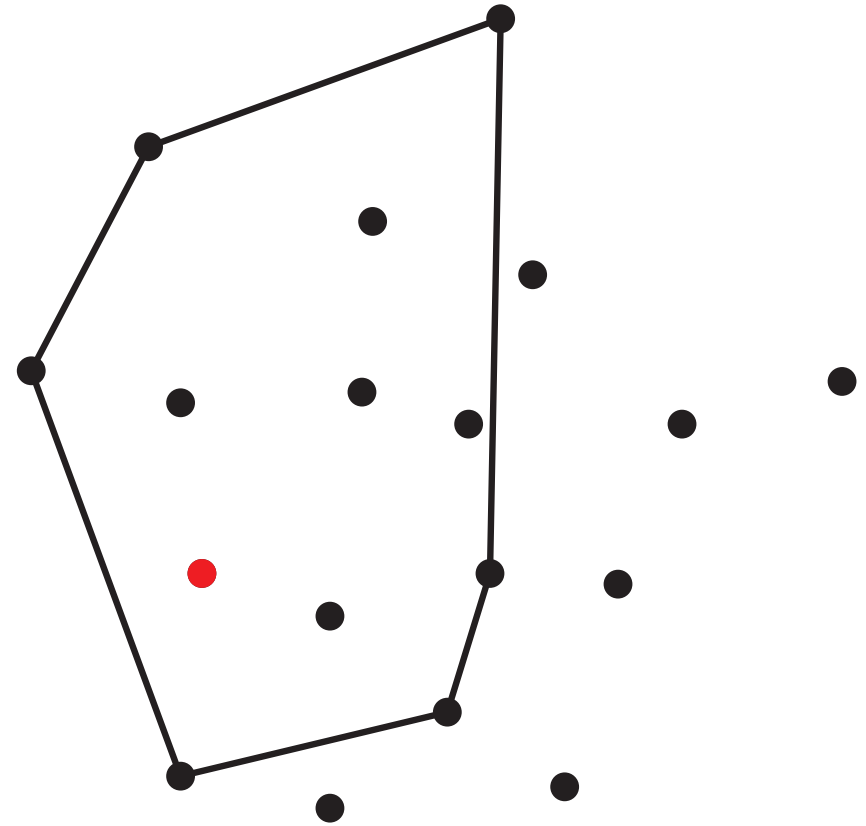
the points  $p_l$  and  $p_r$

the supporting lines

the points  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

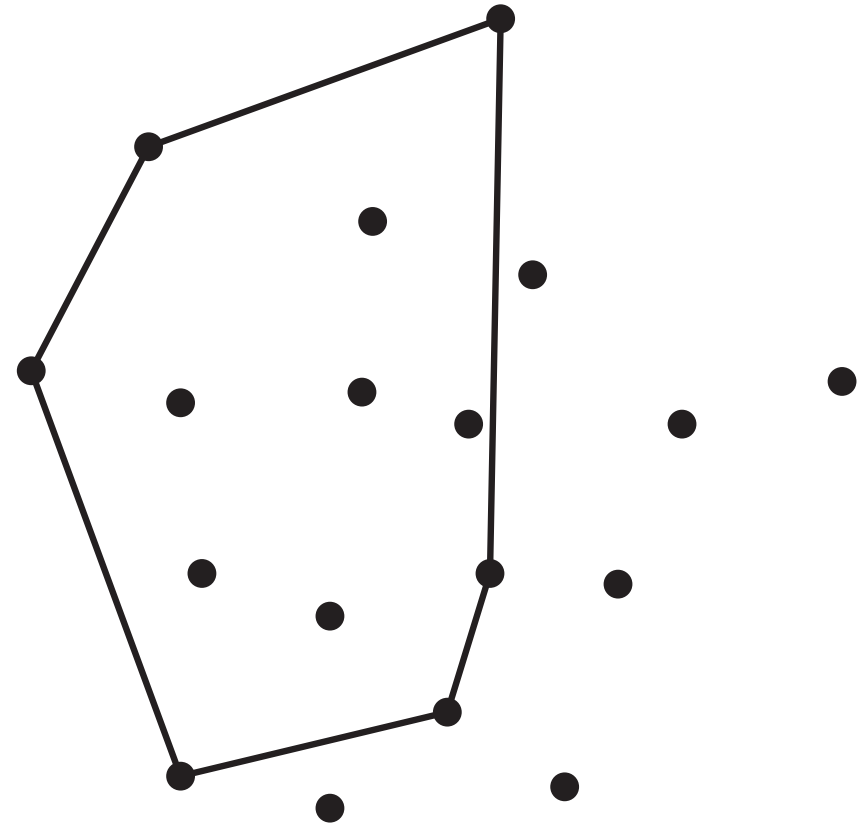
the points  $p_l$  and  $p_r$

the supporting lines

$p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

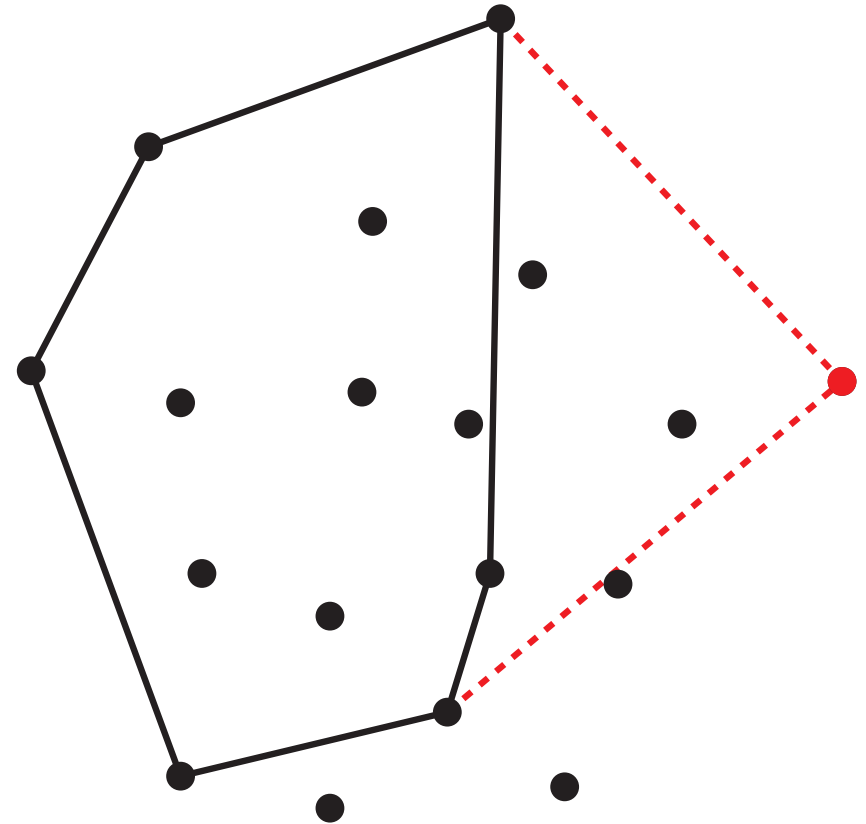
the points  $p_l$  and  $p_r$

the supporting lines

the points  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

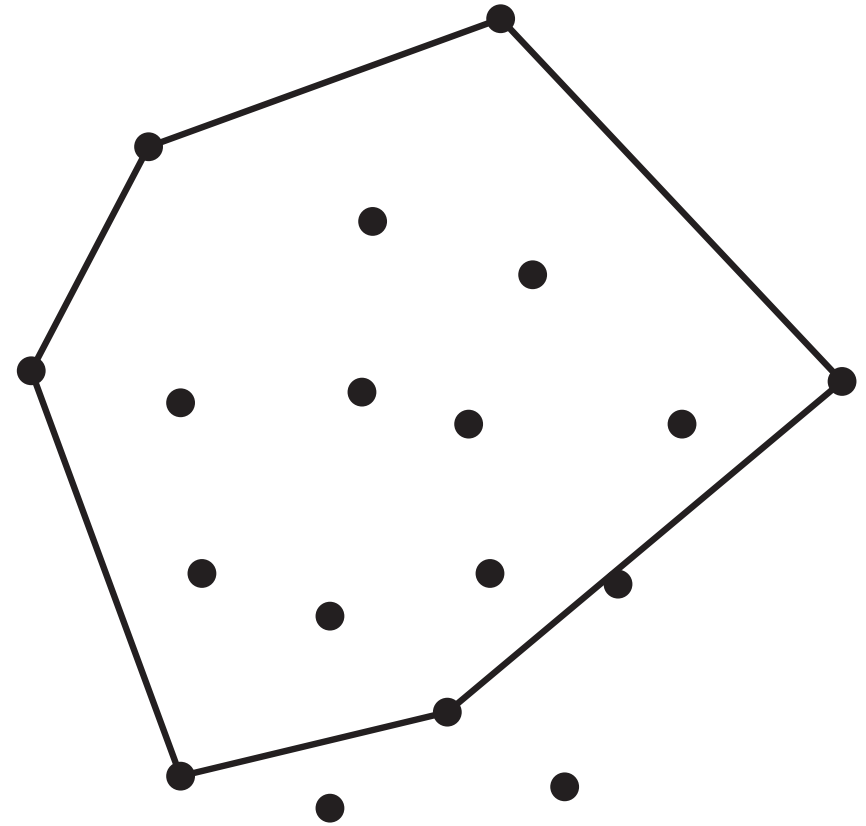
the points  $p_l$  and  $p_r$

the supporting lines

the points  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

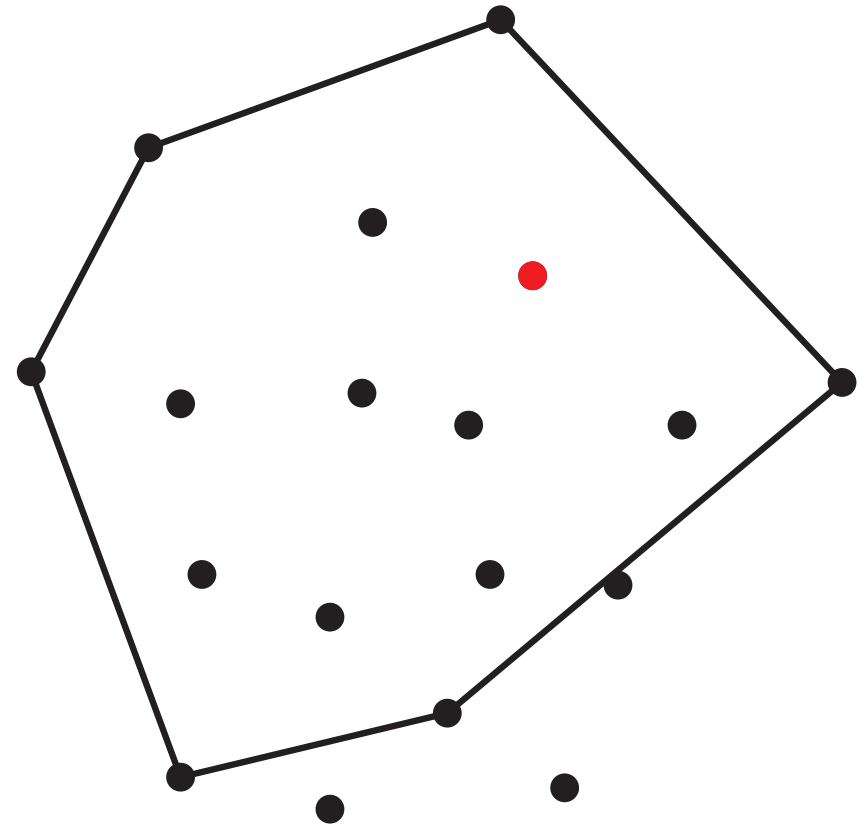
the points  $p_l$  and  $p_r$

the supporting lines

$p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

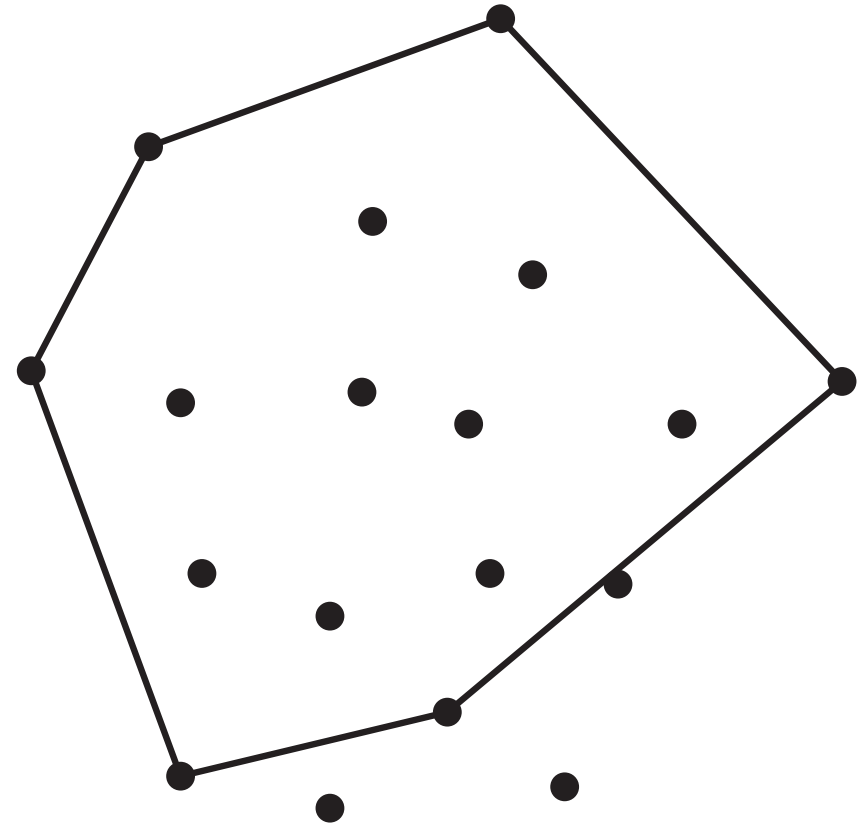
the points  $p_l$  and  $p_r$

the supporting lines

the points  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

do:

the exterior of the polygon defined by  $l$ :

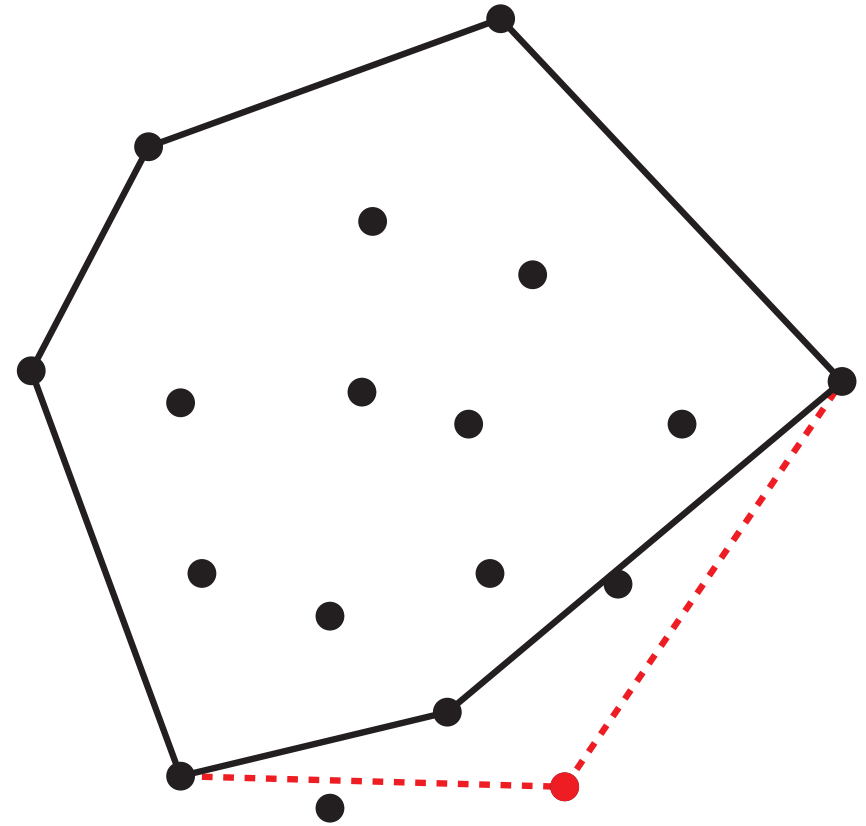
the points  $p_l$  and  $p_r$

the supporting lines

the points  $p_i$  to the polygon

the chain  $p_l, \dots, p_r$  in  $l$

the chain  $p_l, p_i, p_r$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

www.cartagena99.com no se hace responsable de la información contenida en el presente documento en virtud al Artículo 17.1 de la Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico, de 11 de julio de 2002. Si la información contenida en el documento es ilícita o lesiona bienes o derechos de un tercero háganoslo saber y será retirada.

# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

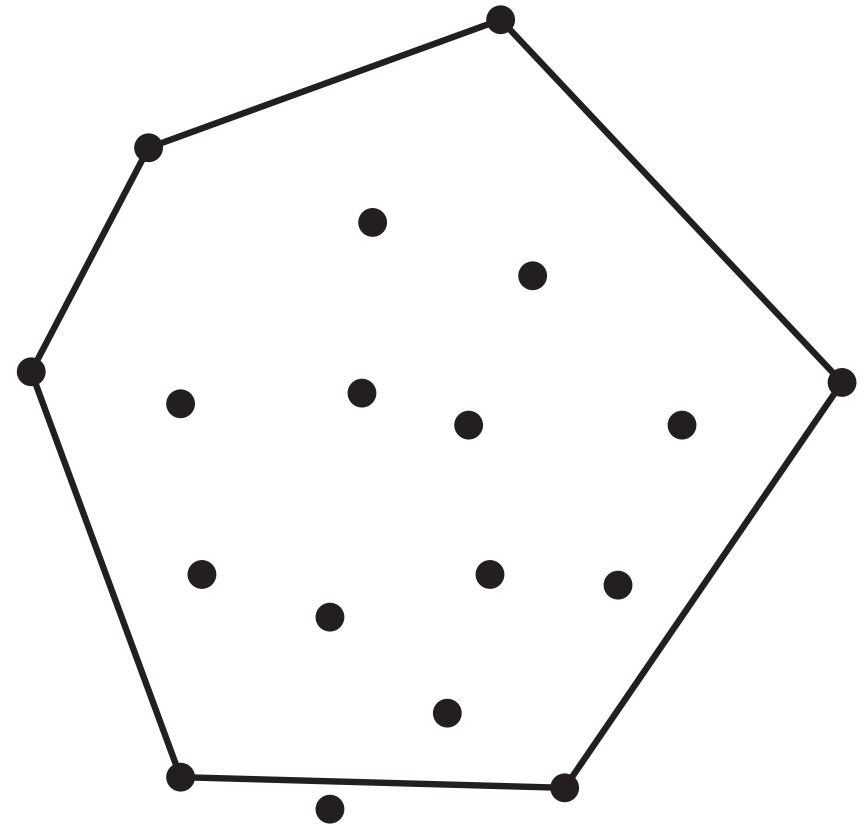
### Advance

From  $i = 4$  to  $n$ , do:

If  $p_i$  lies in the exterior of the polygon defined by  $l$ :

- Compute the points  $p_l$  and  $p_r$  defining the supporting lines from  $p_i$  to the polygon
- Replace the chain  $p_l, \dots, p_r$  in  $l$  with the chain  $p_l, p_i, p_r$

Return  $l$



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

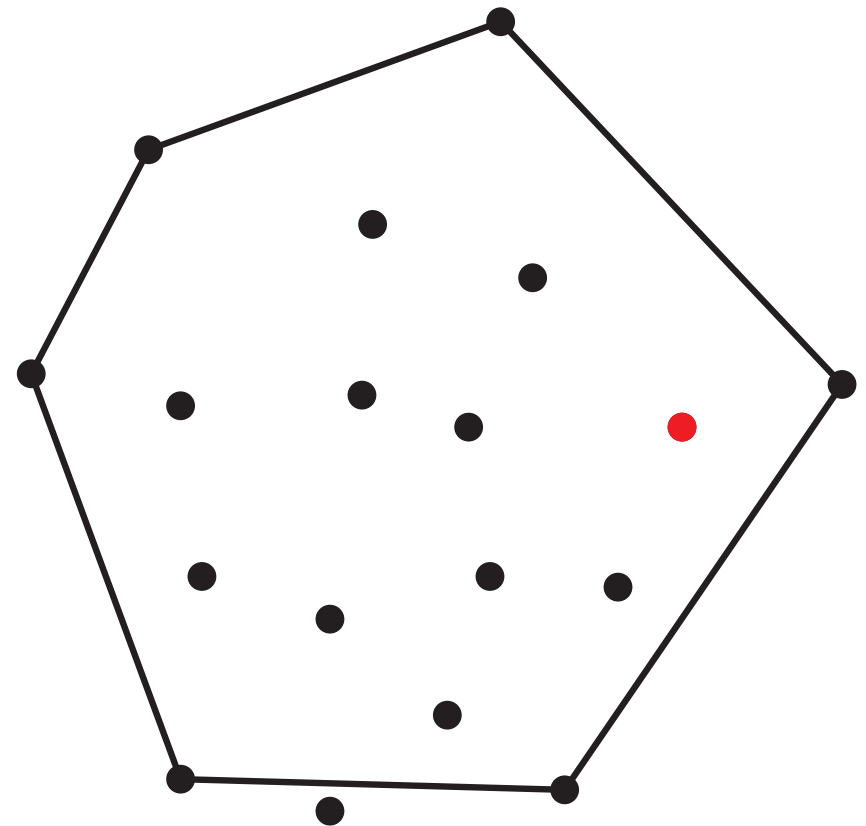
### Advance

From  $i = 4$  to  $n$ , do:

If  $p_i$  lies in the exterior of the polygon defined by  $l$ :

- Compute the points  $p_l$  and  $p_r$  defining the supporting lines from  $p_i$  to the polygon
- Replace the chain  $p_l, \dots, p_r$  in  $l$  with the chain  $p_l, p_i, p_r$

Return  $l$



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

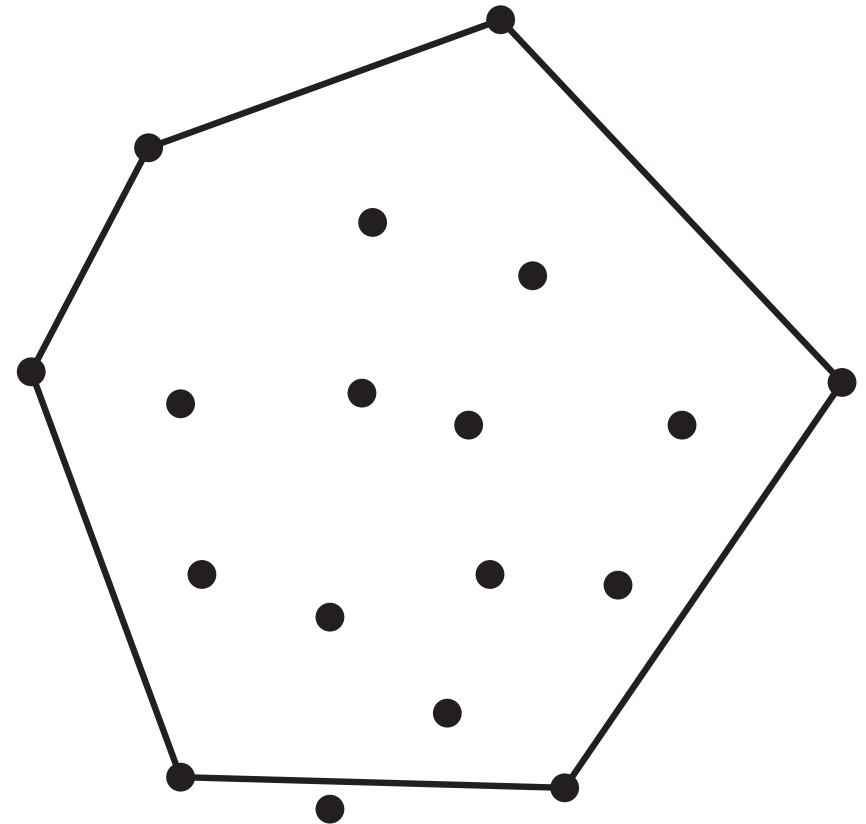
### Advance

From  $i = 4$  to  $n$ , do:

If  $p_i$  lies in the exterior of the polygon defined by  $l$ :

- Compute the points  $p_l$  and  $p_r$  defining the supporting lines from  $p_i$  to the polygon
- Replace the chain  $p_l, \dots, p_r$  in  $l$  with the chain  $p_l, p_i, p_r$

Return  $l$



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

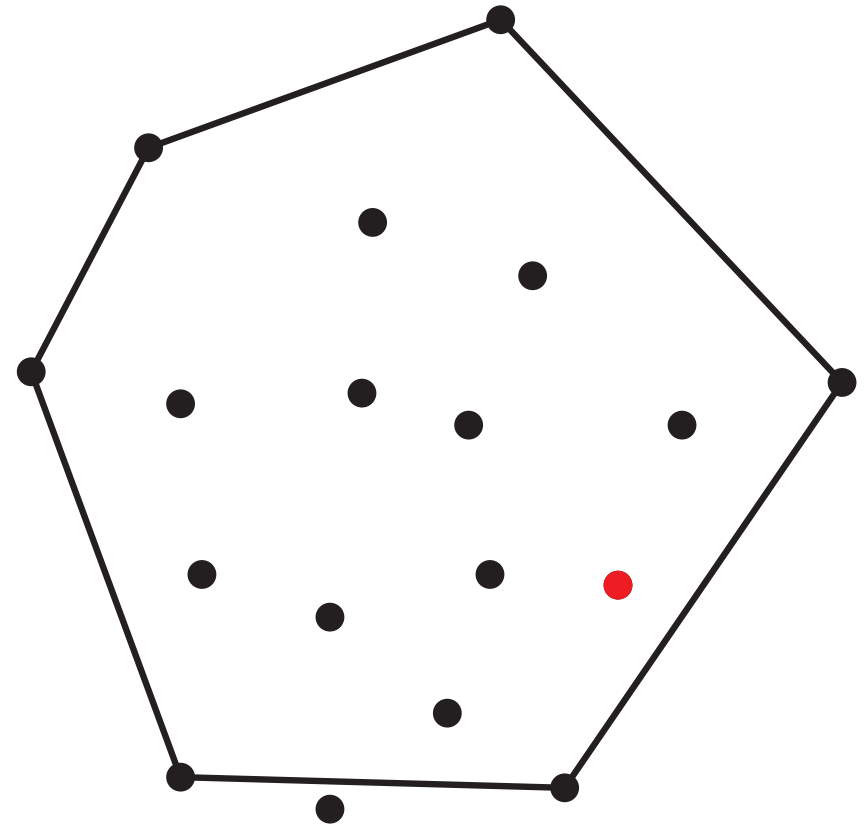
### Advance

From  $i = 4$  to  $n$ , do:

If  $p_i$  lies in the exterior of the polygon defined by  $l$ :

- Compute the points  $p_l$  and  $p_r$  defining the supporting lines from  $p_i$  to the polygon
- Replace the chain  $p_l, \dots, p_r$  in  $l$  with the chain  $p_l, p_i, p_r$

Return  $l$



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

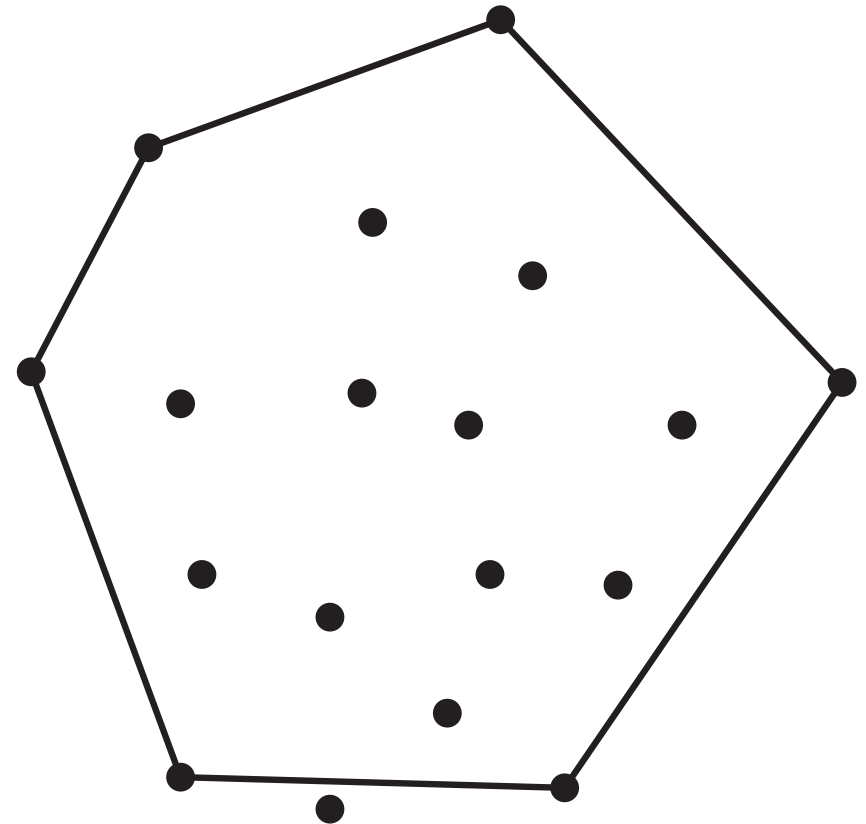
### Advance

From  $i = 4$  to  $n$ , do:

If  $p_i$  lies in the exterior of the polygon defined by  $l$ :

- Compute the points  $p_l$  and  $p_r$  defining the supporting lines from  $p_i$  to the polygon
- Replace the chain  $p_l, \dots, p_r$  in  $l$  with the chain  $p_l, p_i, p_r$

Return  $l$



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

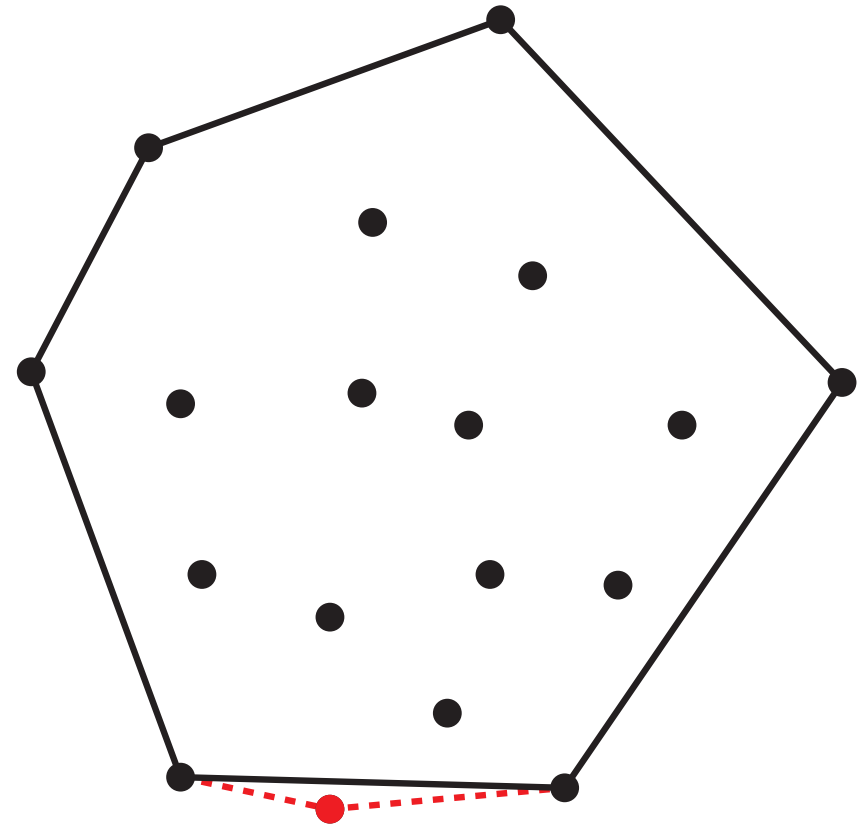
### Advance

From  $i = 4$  to  $n$ , do:

If  $p_i$  lies in the exterior of the polygon defined by  $l$ :

- Compute the points  $p_l$  and  $p_r$  defining the supporting lines from  $p_i$  to the polygon
- Replace the chain  $p_l, \dots, p_r$  in  $l$  with the chain  $p_l, p_i, p_r$

Return  $l$





# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

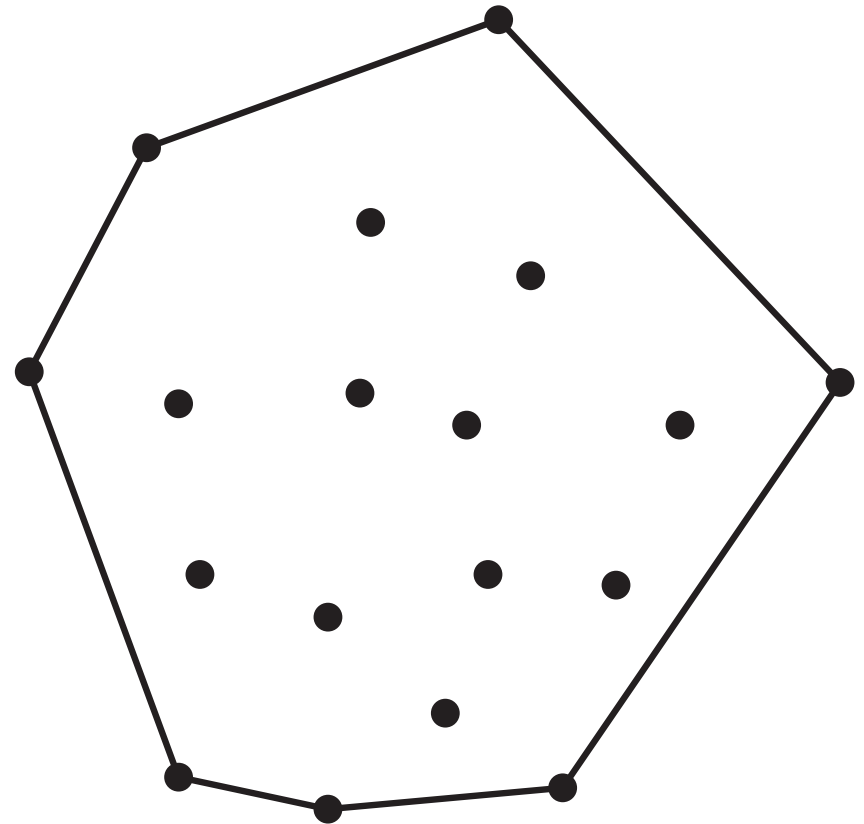
### Advance

From  $i = 4$  to  $n$ , do:

If  $p_i$  lies in the exterior of the polygon defined by  $l$ :

- Compute the points  $p_l$  and  $p_r$  defining the supporting lines from  $p_i$  to the polygon
- Replace the chain  $p_l, \dots, p_r$  in  $l$  with the chain  $p_l, p_i, p_r$

Return  $l$



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

### Advance

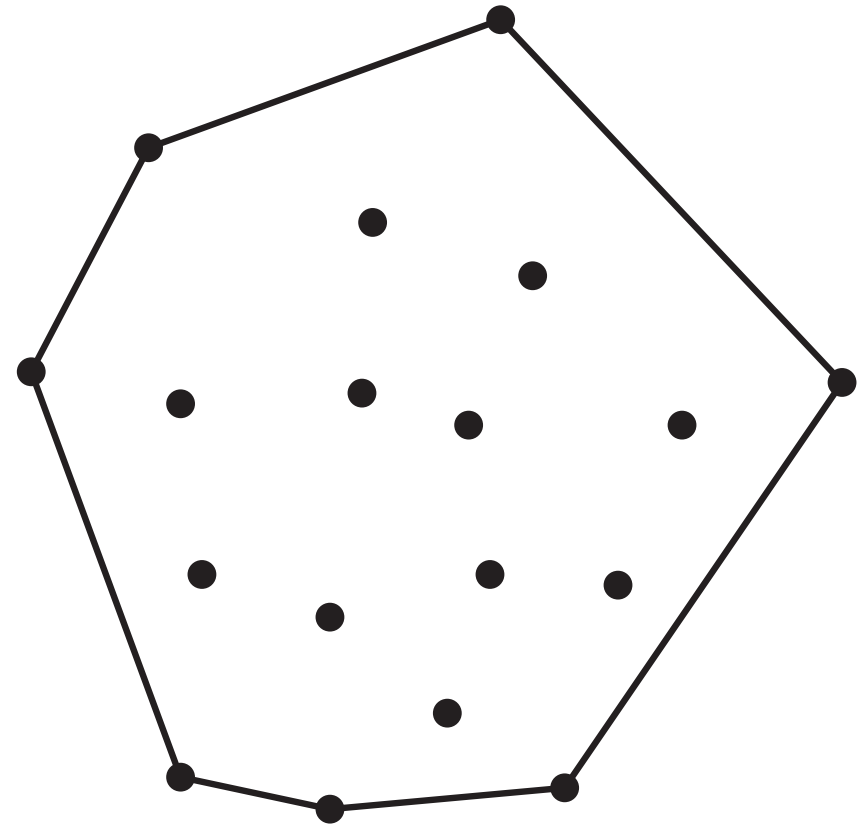
From  $i = 4$  to  $n$ , do:

If  $p_i$  lies in the exterior of the polygon defined by  $l$ :

- Compute the points  $p_l$  and  $p_r$  defining the supporting lines from  $p_i$  to the polygon
- Replace the chain  $p_l, \dots, p_r$  in  $l$  with the chain  $p_l, p_i, p_r$

Return  $l$

**Running time:**  $O(n \log n)$



# CONVEX HULL IN 2D

## Incremental algorithm

### Initialization

$$l = p_1, p_2, p_3$$

### Advance

From  $i = 4$  to  $n$ , do:

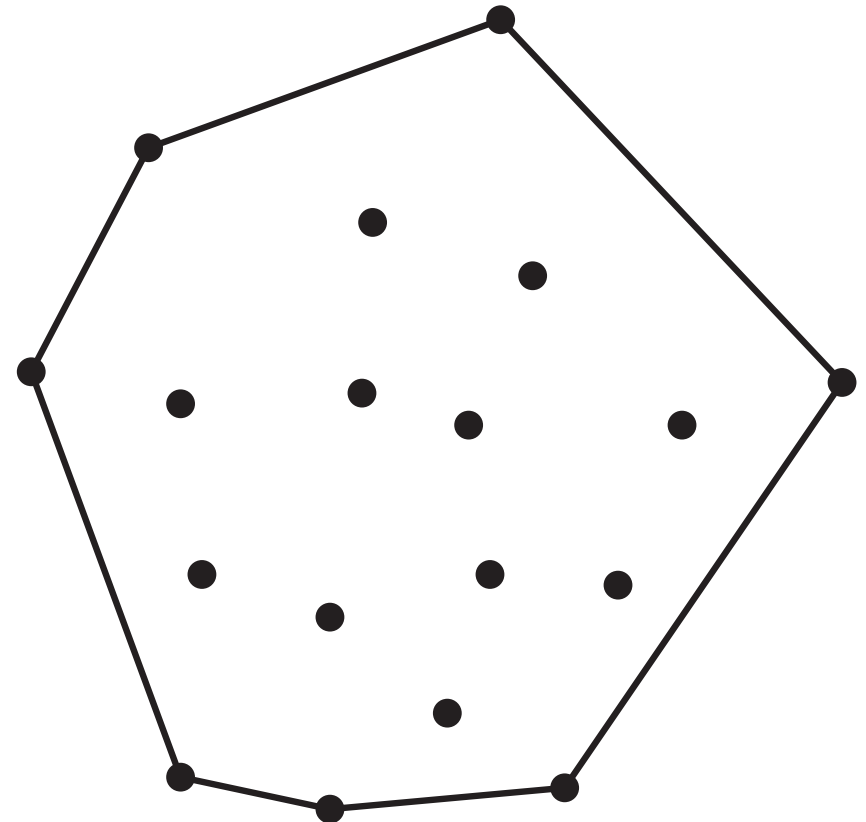
If  $p_i$  lies in the exterior of the polygon defined by  $l$ :

- Compute the points  $p_l$  and  $p_r$  defining the supporting lines from  $p_i$  to the polygon
- Replace the chain  $p_l, \dots, p_r$  in  $l$  with the chain  $p_l, p_i, p_r$

Return  $l$

**Running time:**  $O(n \log n)$

By storing  $l$  in a structure allowing binary search and updatings (insertions and deletions) in  $O(\log n)$  time.



# CONVEX HULL IN 2D

Divide-and-conquer algorithm

# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

### Division

1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae

# CONVEX HULL IN 2D

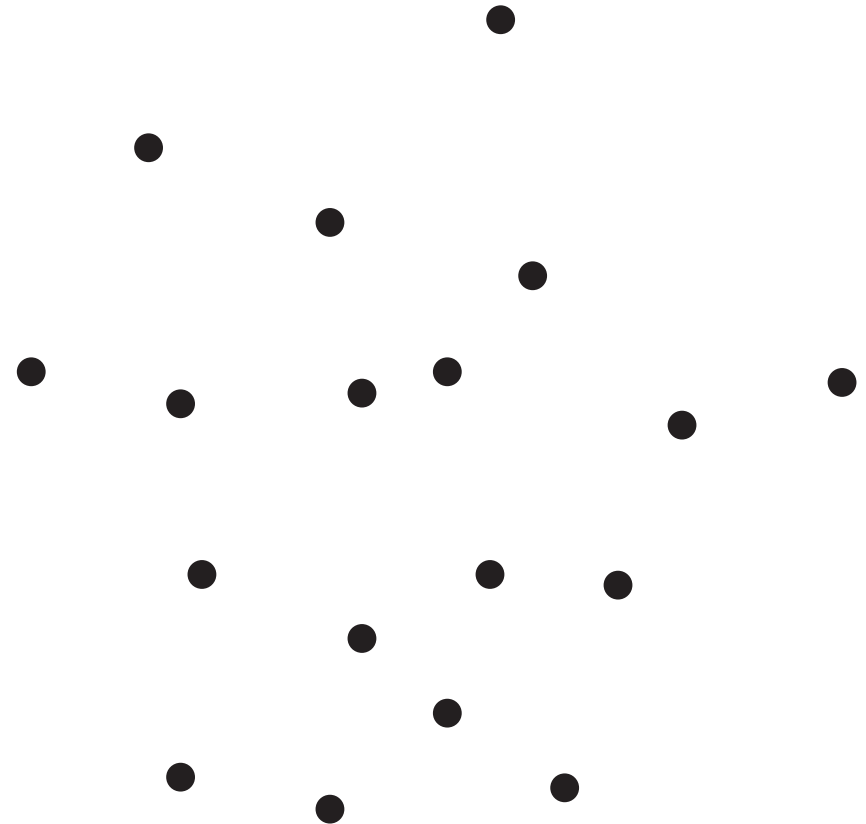
## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

### Division

1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae



# CONVEX HULL IN 2D

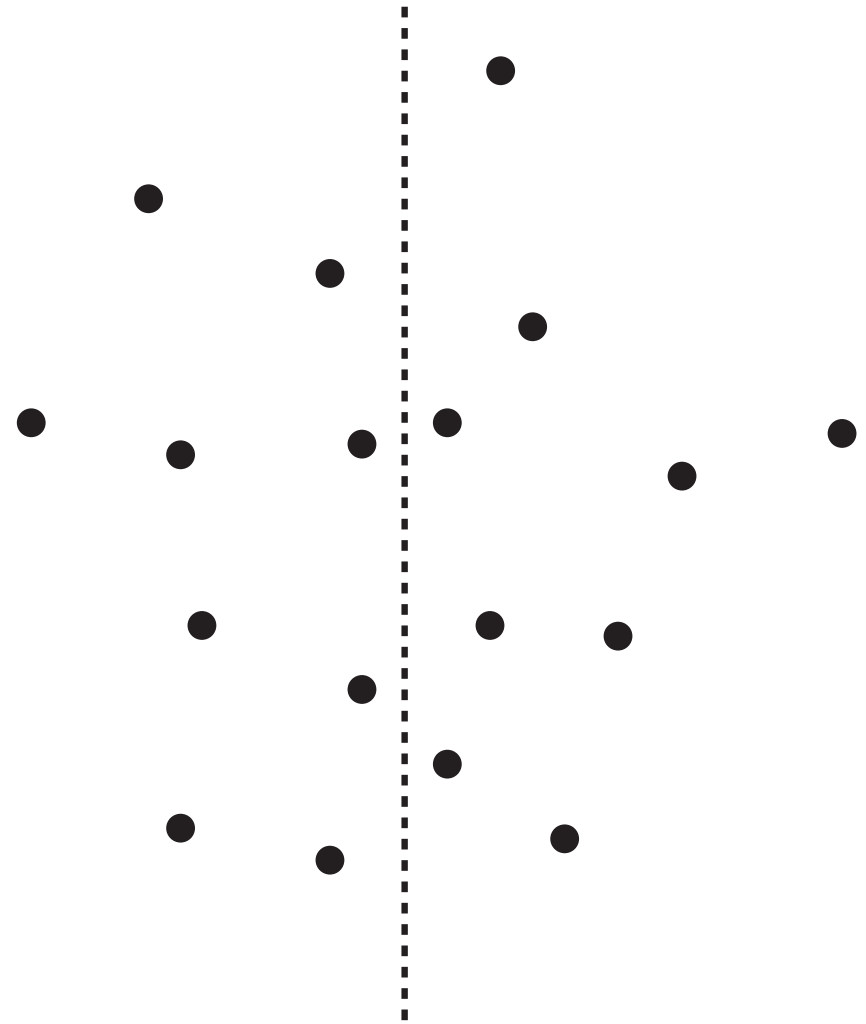
## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

### Division

1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae





# CONVEX HULL IN 2D

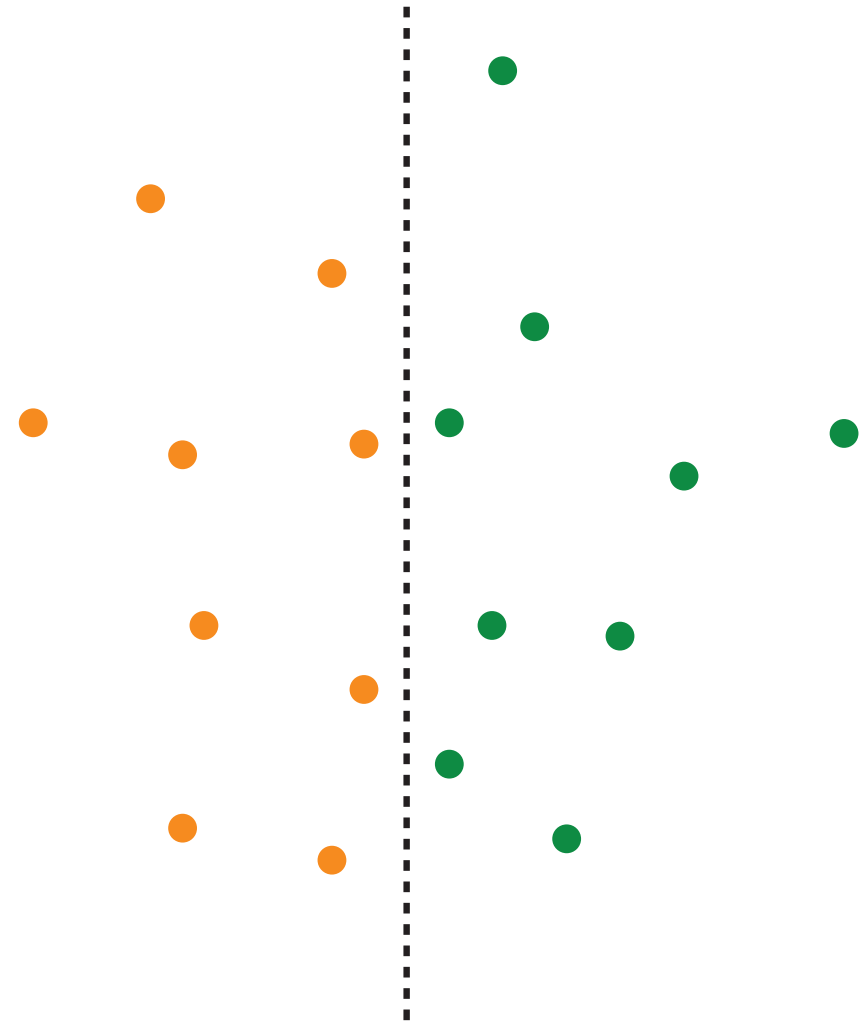
## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

### Division

1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae



# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

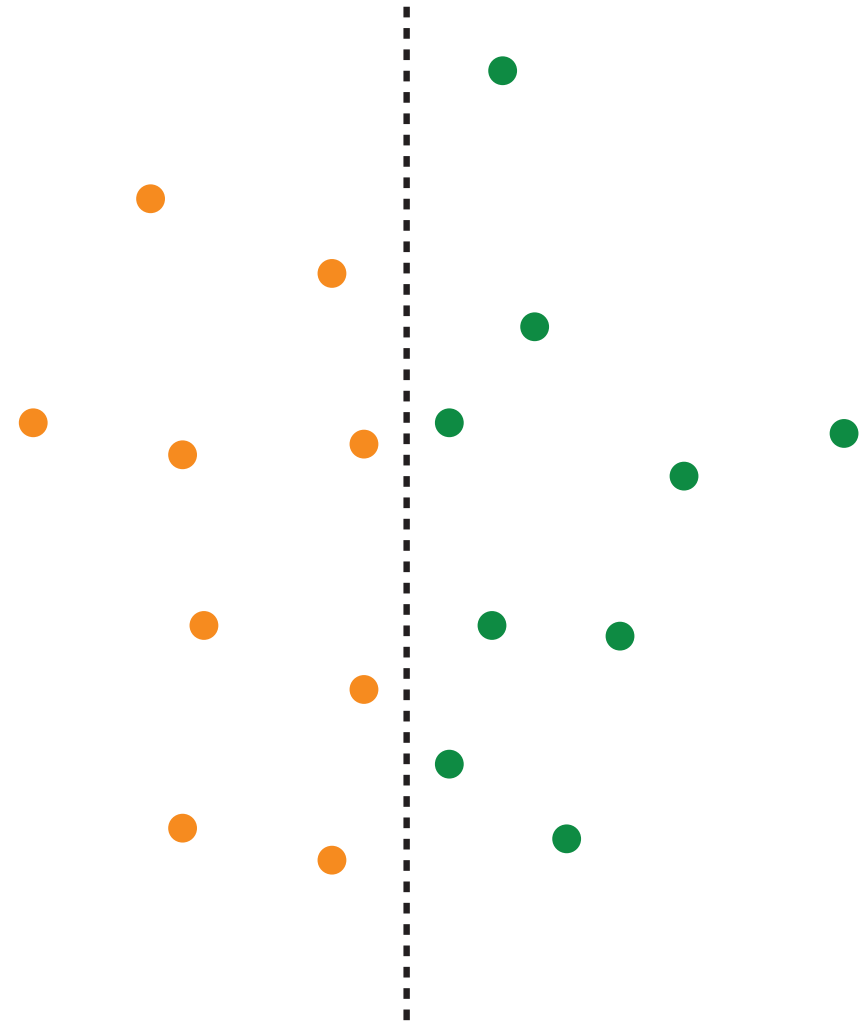
1. Sort the points by abscissae

### Division

1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae

### Recursion

1. Recursively compute the convex hull of the two subsets



# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

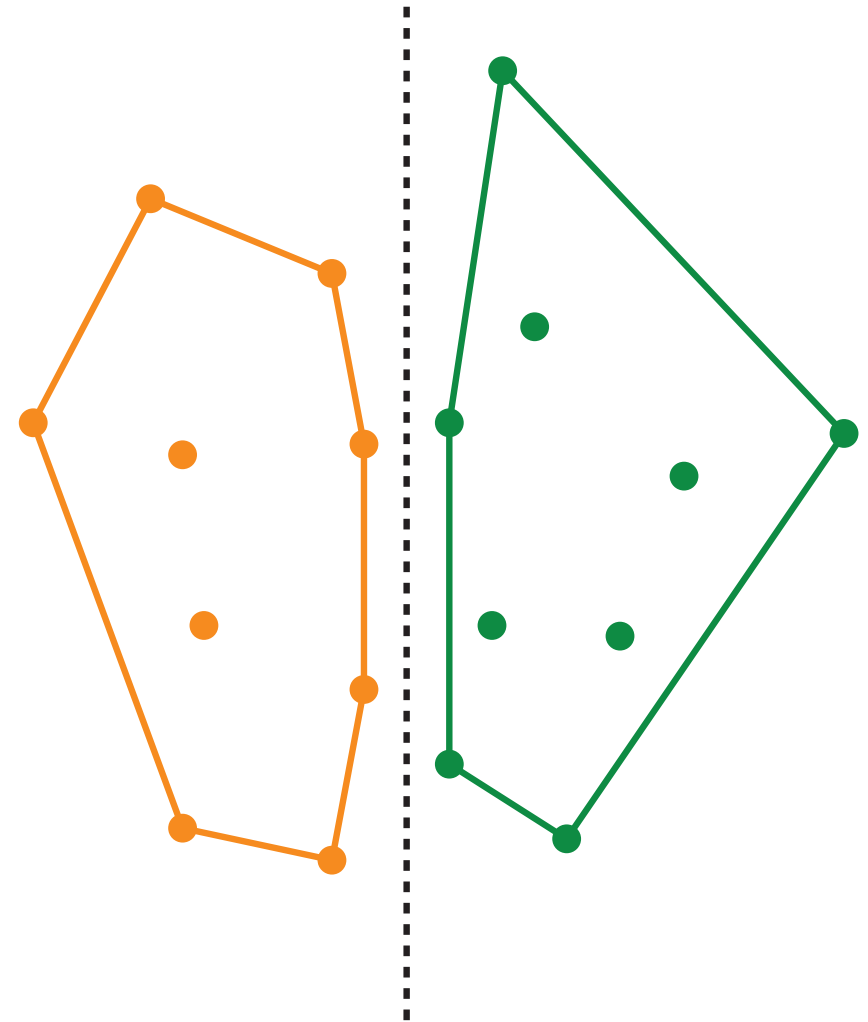
1. Sort the points by abscissae

### Division

1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae

### Recursion

1. Recursively compute the convex hull of the two subsets



# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

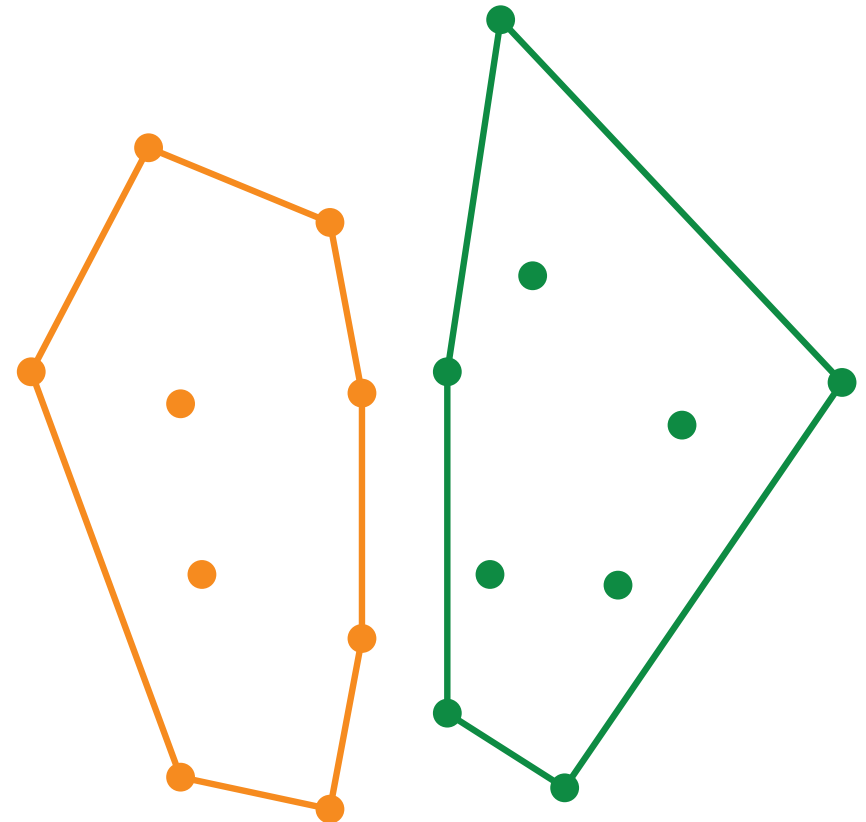
1. Sort the points by abscissae

### Division

1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae

### Recursion

1. Recursively compute the convex hull of the two subsets



# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

### Division

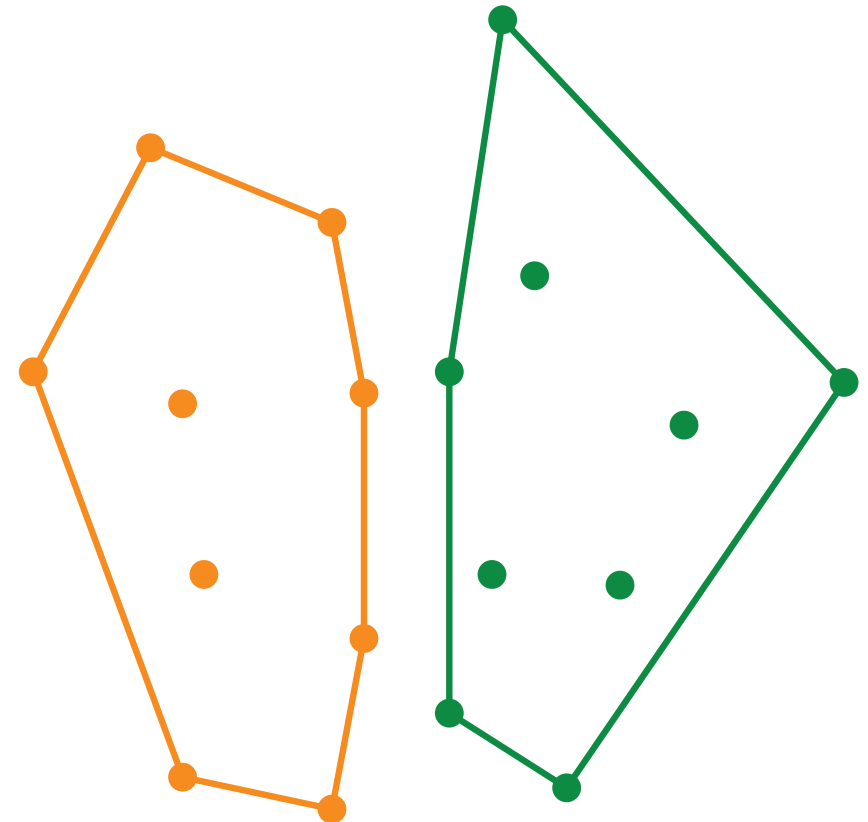
1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae

### Recursion

1. Recursively compute the convex hull of the two subsets

### Merge

1. Compute the external common tangents of the two convex polygons
2. Delete the interior chains of the two polygons and join the external chains through the supporting segments



# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

### Division

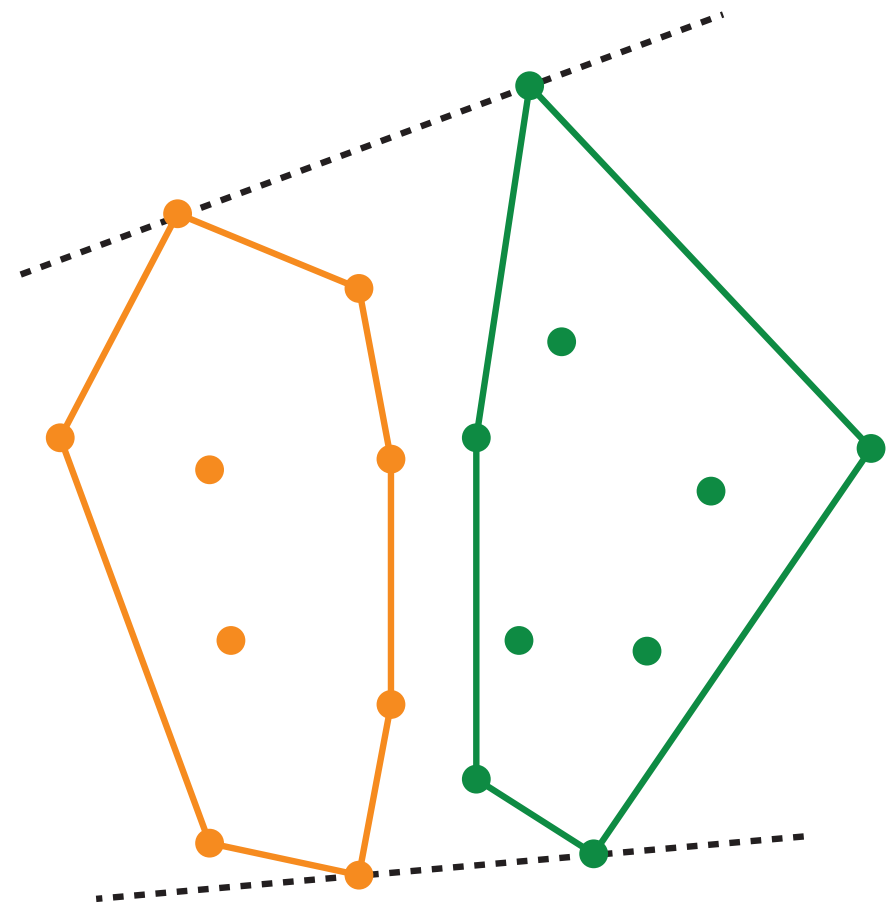
1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae

### Recursion

1. Recursively compute the convex hull of the two subsets

### Merge

1. Compute the external common tangents of the two convex polygons
2. Delete the interior chains of the two polygons and join the external chains through the supporting segments



# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

### Division

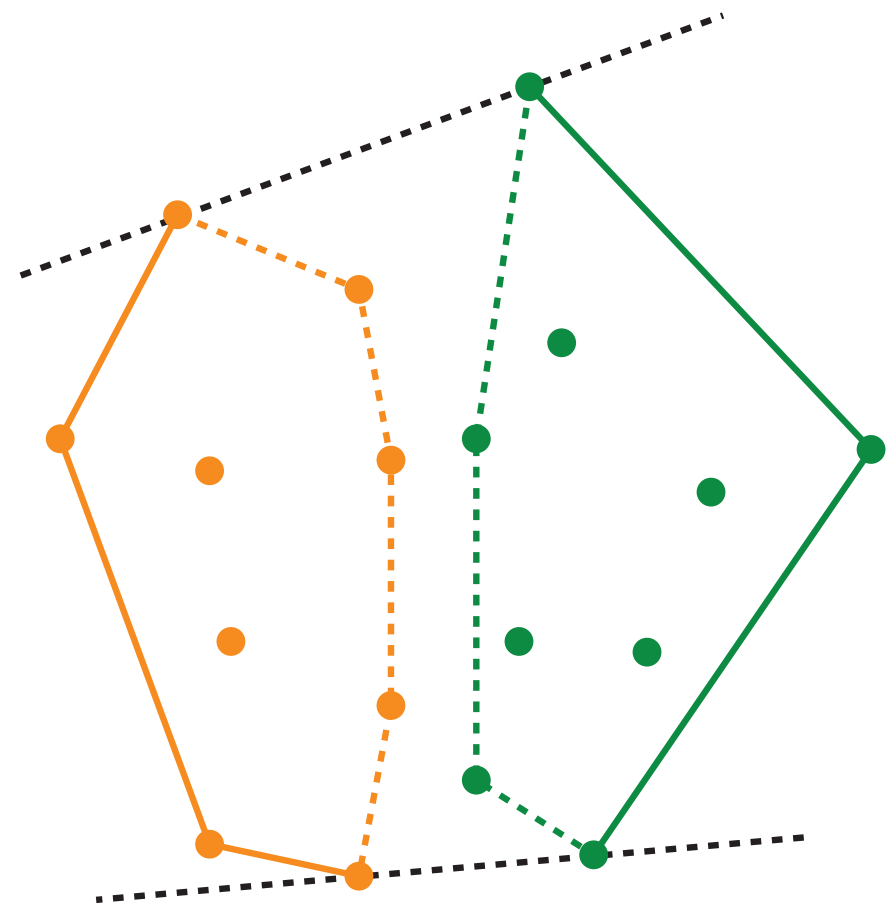
1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae

### Recursion

1. Recursively compute the convex hull of the two subsets

### Merge

1. Compute the external common tangents of the two convex polygons
2. Delete the interior chains of the two polygons and join the external chains through the supporting segments



# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

### Division

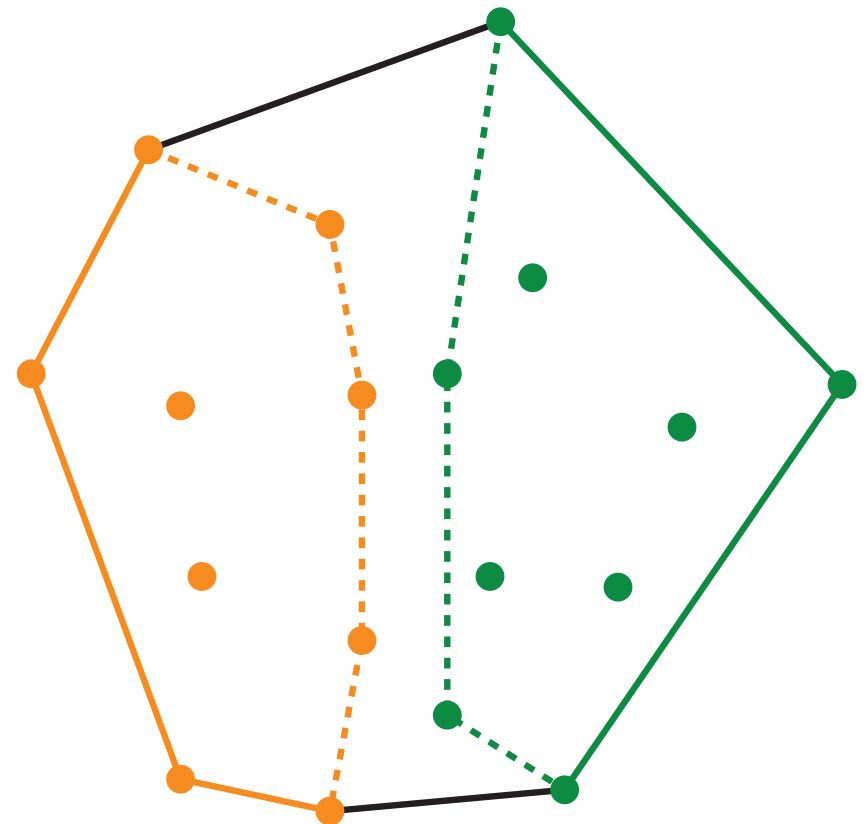
1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae

### Recursion

1. Recursively compute the convex hull of the two subsets

### Merge

1. Compute the external common tangents of the two convex polygons
2. Delete the interior chains of the two polygons and join the external chains through the supporting segments





# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Initialization

1. Sort the points by abscissae

### Division

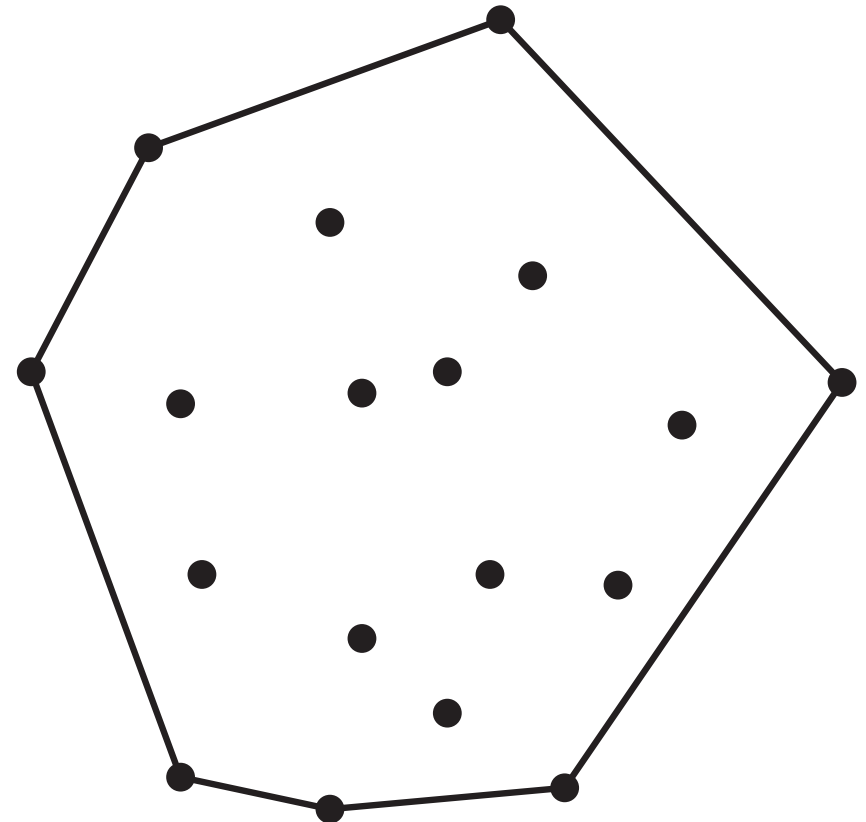
1. Divide the points  $(x_i, y_i)$  into two subsets, wrt the median value of the abscissae

### Recursion

1. Recursively compute the convex hull of the two subsets

### Merge

1. Compute the external common tangents of the two convex polygons
2. Delete the interior chains of the two polygons and join the external chains through the supporting segments

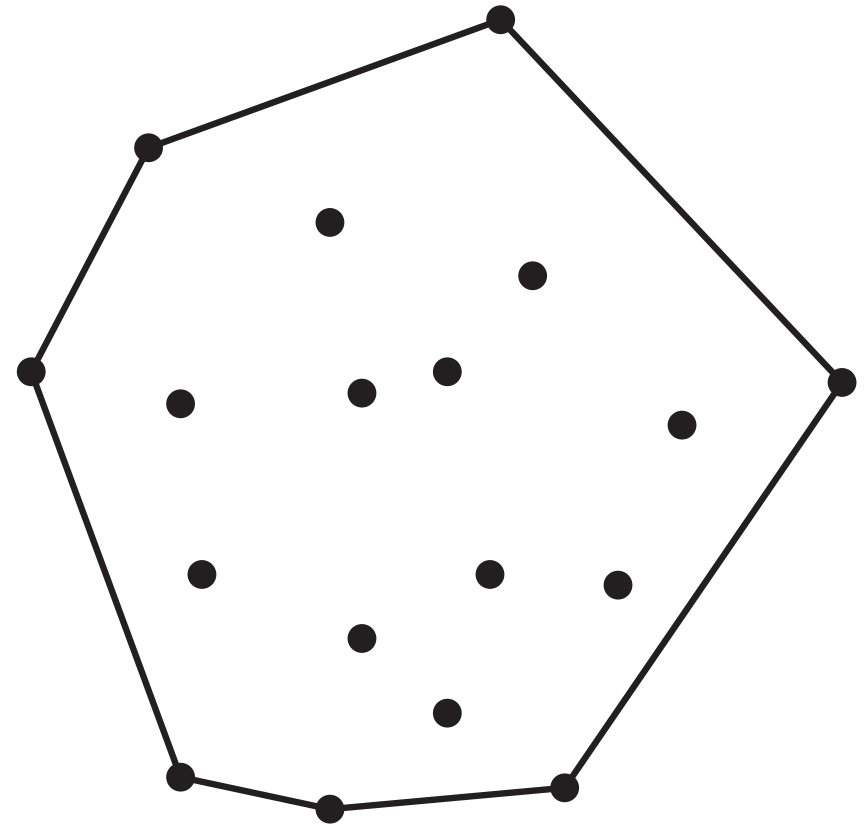


# CONVEX HULL IN 2D

Divide-and-conquer algorithm

Running time

Initialization:  $O(n \log n)$  (only once)



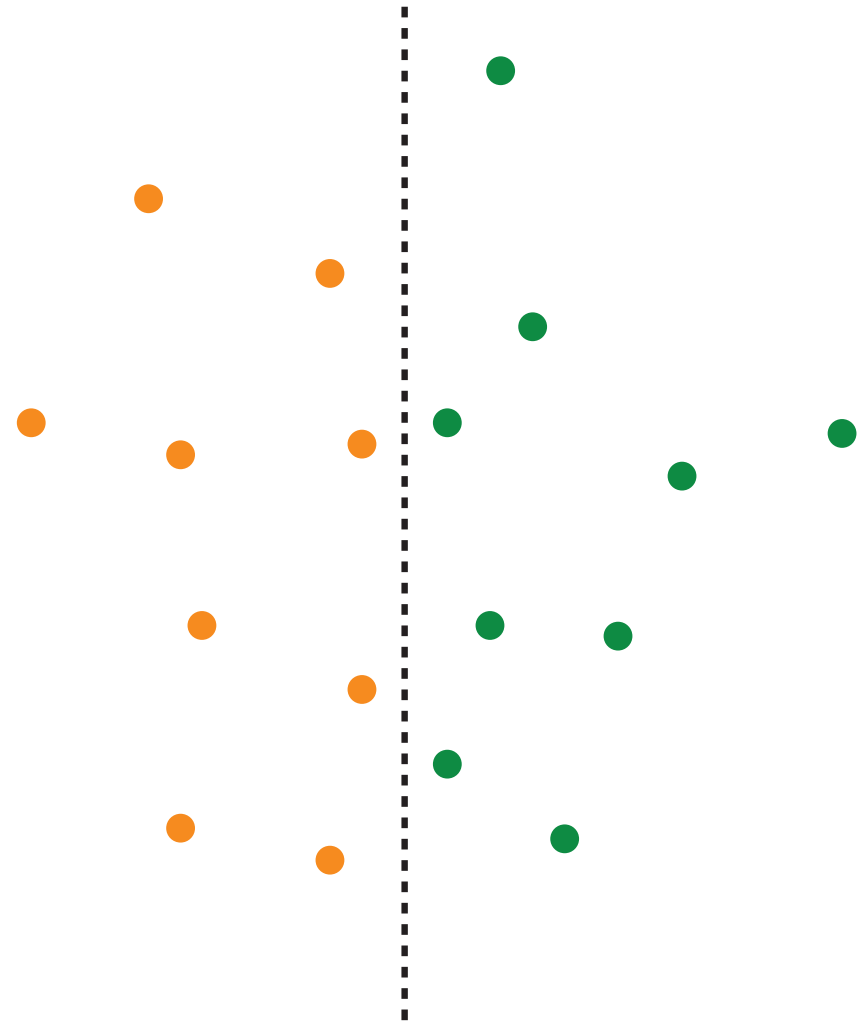
# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Running time

Initialization:  $O(n \log n)$  (only once)

Division:  $O(n)$



# CONVEX HULL IN 2D

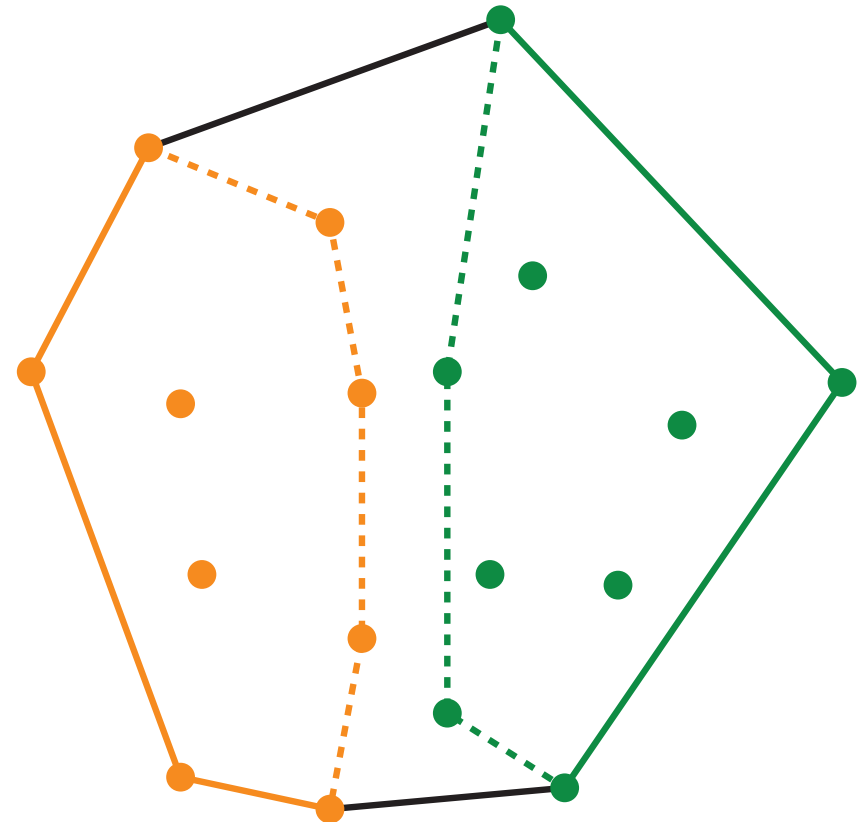
## Divide-and-conquer algorithm

### Running time

Initialization:  $O(n \log n)$  (only once)

Division:  $O(n)$

Merge:  $O(n)$



# CONVEX HULL IN 2D

## Divide-and-conquer algorithm

### Running time

Initialization:  $O(n \log n)$  (only once)

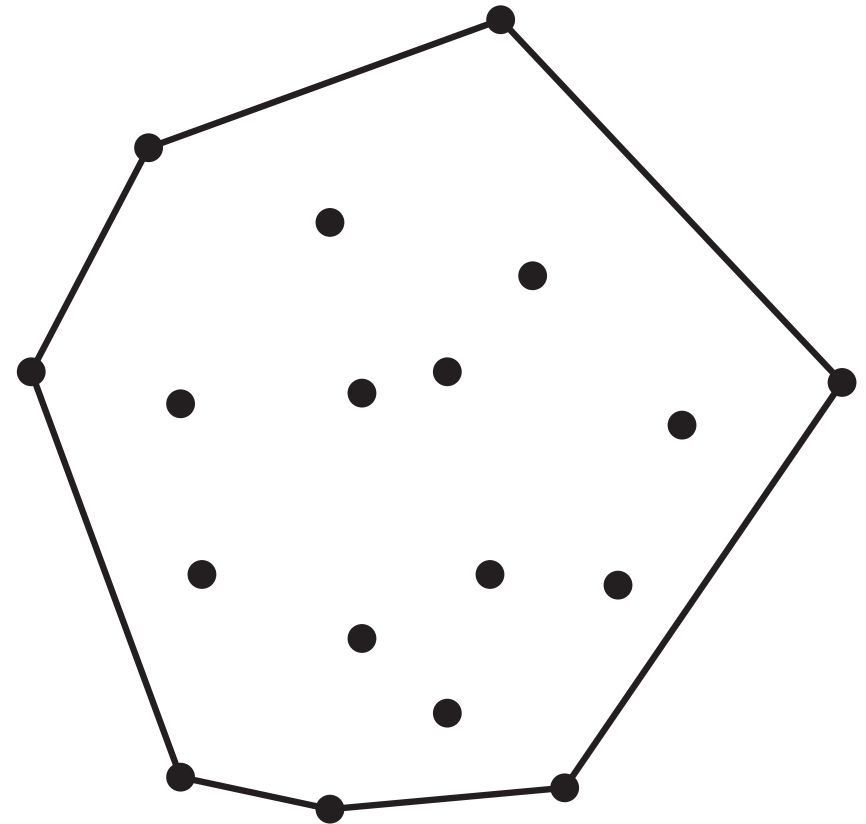
Division:  $O(n)$

Merge:  $O(n)$

Advance:

$$T(n) = 2T\left(\frac{n}{2}\right) + O(n) = O(n \log n)$$

Overall:  $O(n \log n)$



# CONVEX HULL IN 2D

Lower bound

# CONVEX HULL IN 2D

## Lower bound

**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers

# CONVEX HULL IN 2D

## Lower bound

**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers





# CONVEX HULL IN 2D

## Lower bound

**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers



**Input:**  $n$  points

$p_1, \dots, p_n$ , with  $p_i = (x_i, x_i^2)$



# CONVEX HULL IN 2D

## Lower bound

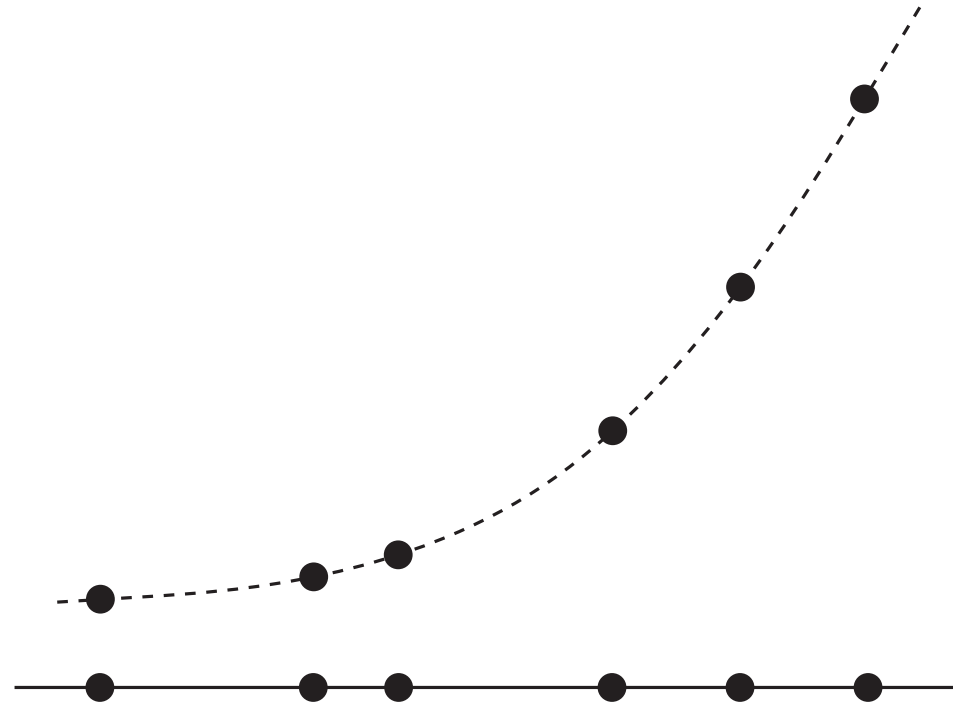
**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers



**Input:**  $n$  points

$p_1, \dots, p_n$ , with  $p_i = (x_i, x_i^2)$



# CONVEX HULL IN 2D

## Lower bound

**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers



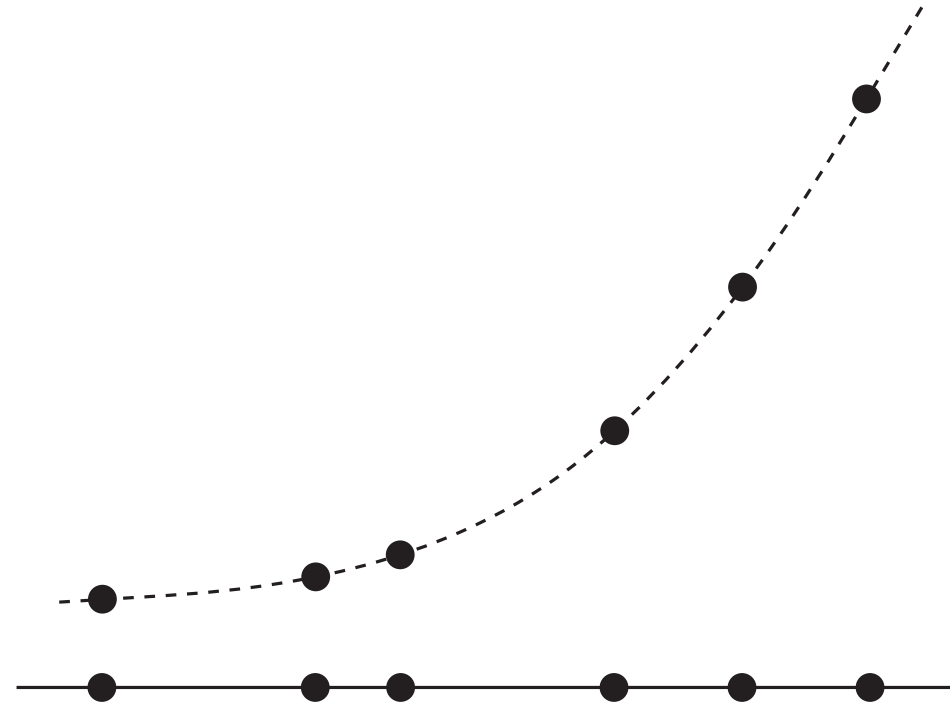
**Input:**  $n$  points

$p_1, \dots, p_n$ , with  $p_i = (x_i, x_i^2)$



**Output:** convex hull of the points

Sorted list of the vertices of the convex hull



# CONVEX HULL IN 2D

## Lower bound

**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers



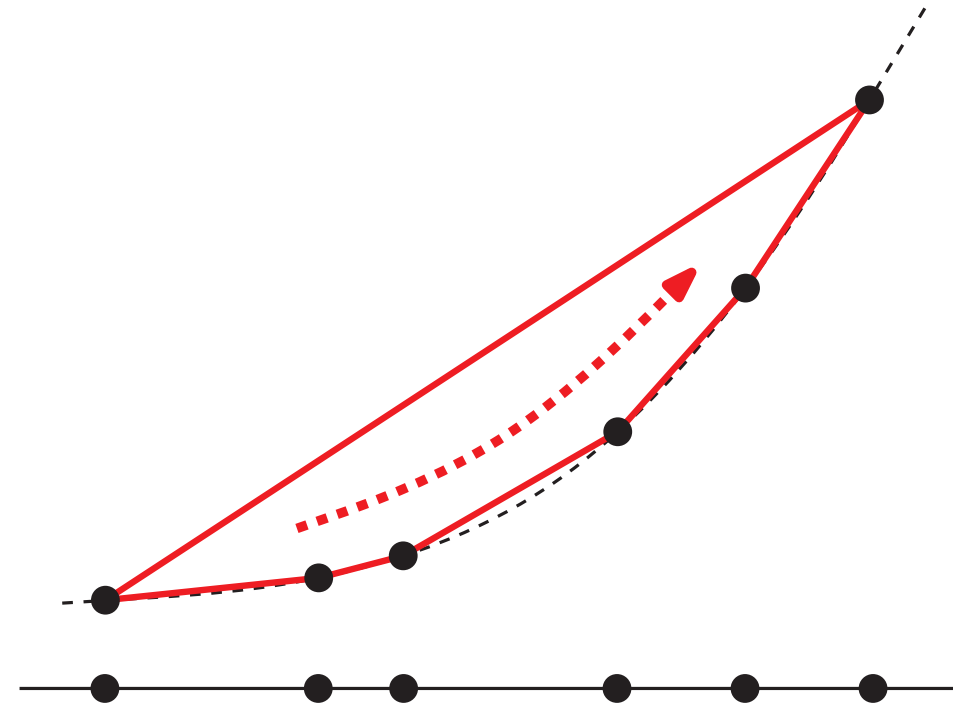
**Input:**  $n$  points

$p_1, \dots, p_n$ , with  $p_i = (x_i, x_i^2)$



**Output:** convex hull of the points

Sorted list of the vertices of the convex hull



# CONVEX HULL IN 2D

## Lower bound

**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers



**Input:**  $n$  points

$p_1, \dots, p_n$ , with  $p_i = (x_i, x_i^2)$



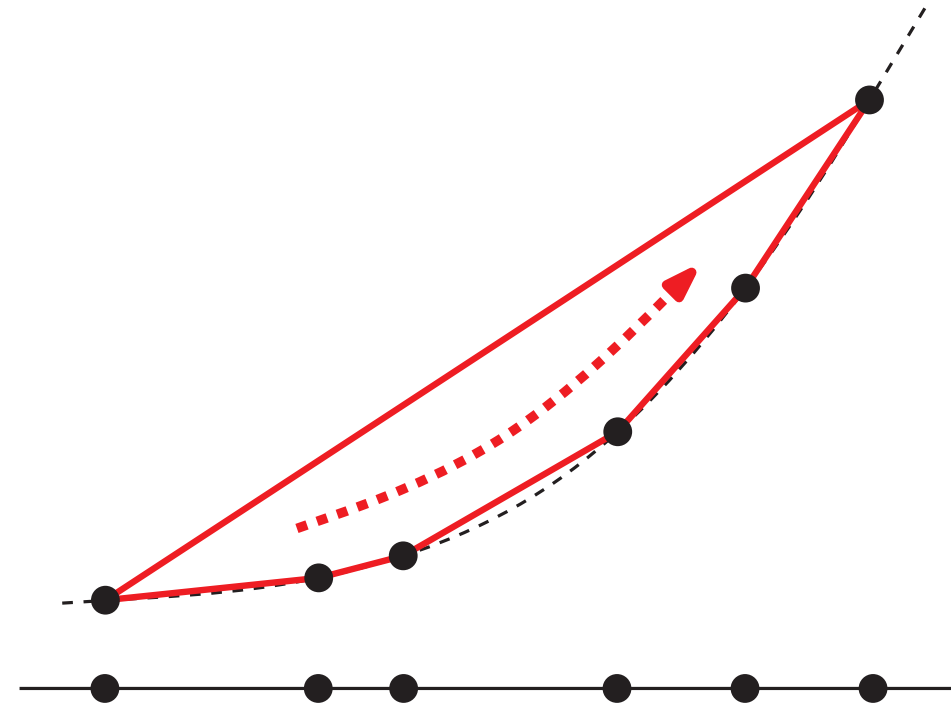
**Output:** convex hull of the points

Sorted list of the vertices of the convex hull



**Output:** sorting the numbers

Sorted list of the numbers  $x_1, \dots, x_n$



# CONVEX HULL IN 2D

## Lower bound

**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers



**Input:**  $n$  points

$p_1, \dots, p_n$ , with  $p_i = (x_i, x_i^2)$



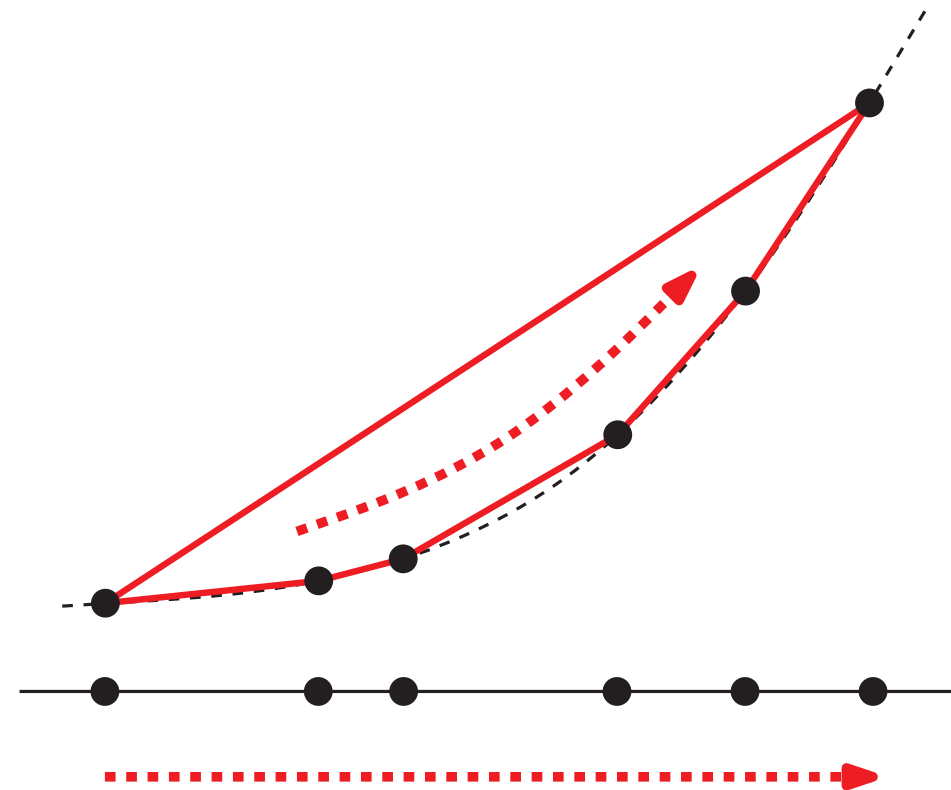
**Output:** convex hull of the points

Sorted list of the vertices of the convex hull



**Output:** sorting the numbers

Sorted list of the numbers  $x_1, \dots, x_n$

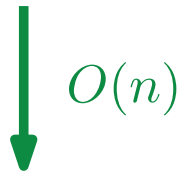


# CONVEX HULL IN 2D

## Lower bound

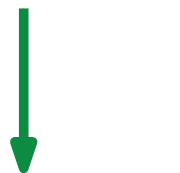
**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers



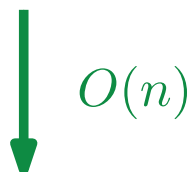
**Input:**  $n$  points

$p_1, \dots, p_n$ , with  $p_i = (x_i, x_i^2)$



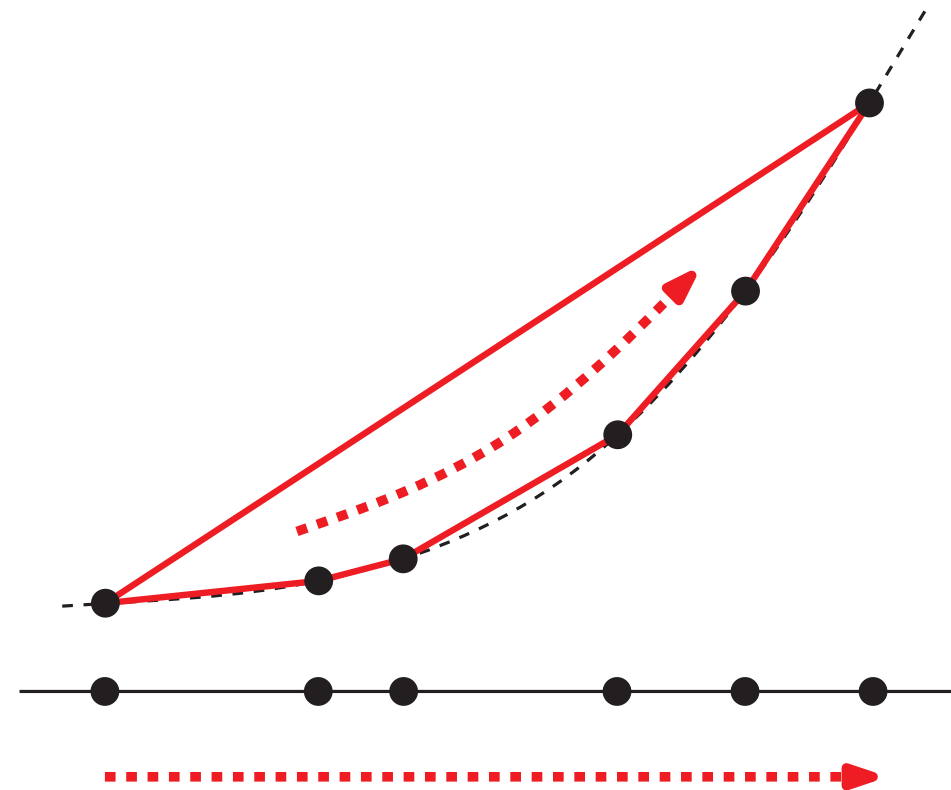
**Output:** convex hull of the points

Sorted list of the vertices of the convex hull



**Output:** sorting the numbers

Sorted list of the numbers  $x_1, \dots, x_n$



# CONVEX HULL IN 2D

## Lower bound

**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers

$O(n)$

**Input:**  $n$  points

$p_1, \dots, p_n$ , with  $p_i = (x_i, x_i^2)$

**Output:** convex hull of the points

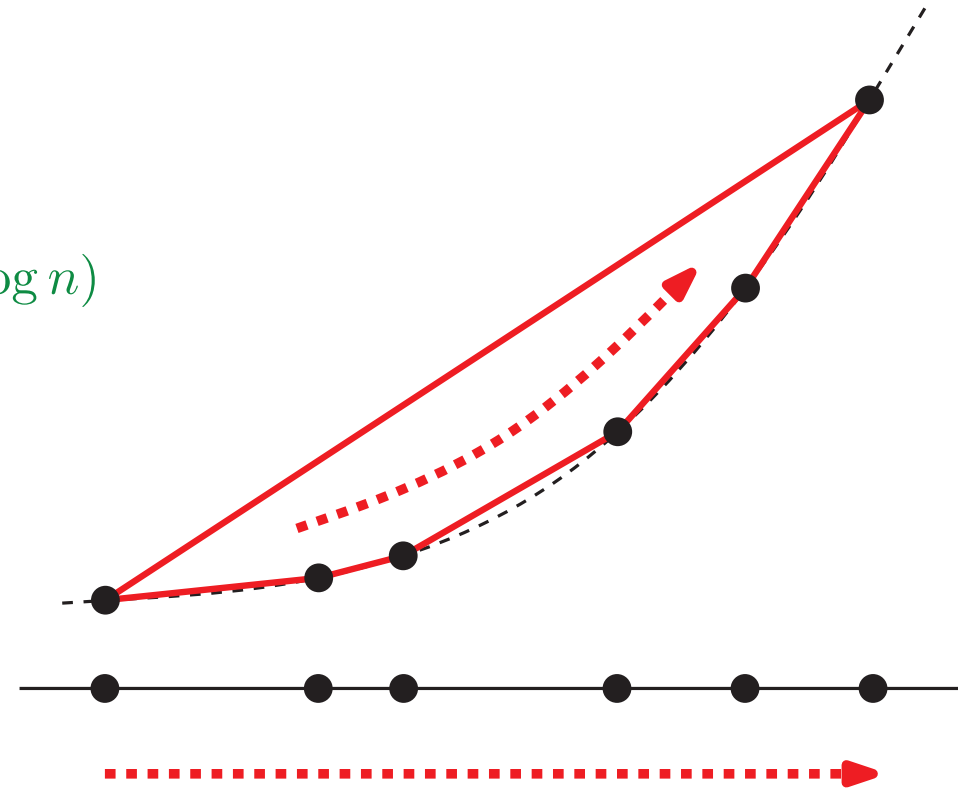
Sorted list of the vertices of the convex hull

$O(n)$

**Output:** sorting the numbers

Sorted list of the numbers  $x_1, \dots, x_n$

$\Omega(n \log n)$





# CONVEX HULL IN 2D

## Lower bound

**Input:**  $n$  real numbers

$x_1, \dots, x_n$  real numbers

$O(n)$

**Input:**  $n$  points

$p_1, \dots, p_n$ , with  $p_i = (x_i, x_i^2)$

$\Omega(n \log n)$

**Output:** convex hull of the points

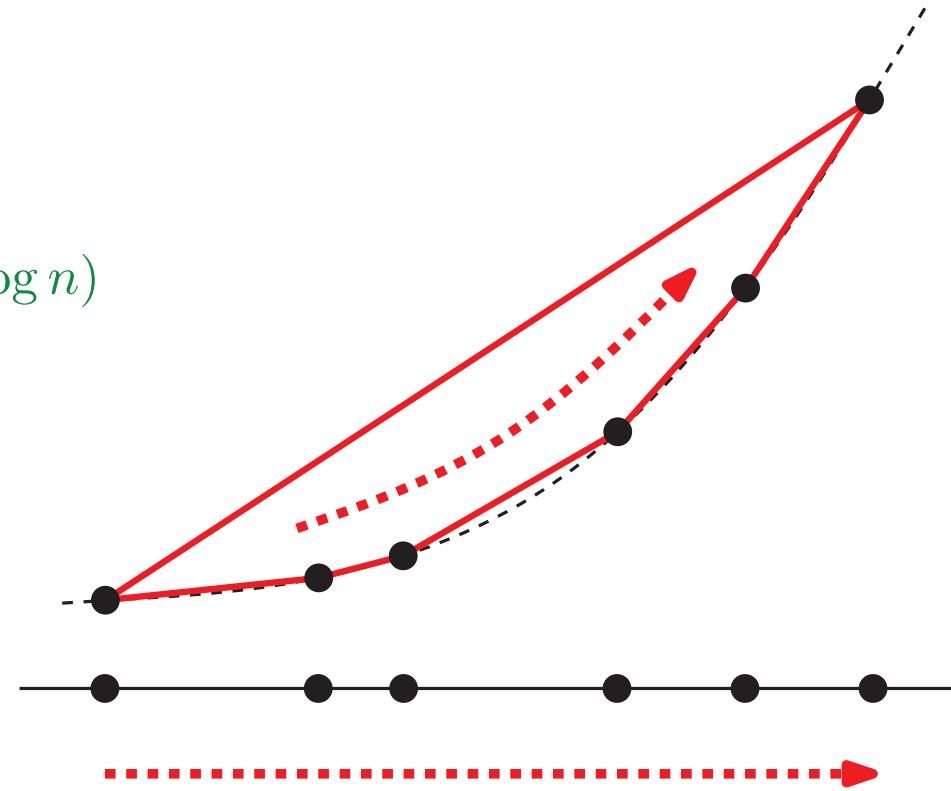
Sorted list of the vertices of the convex hull

$O(n)$

**Output:** sorting the numbers

Sorted list of the numbers  $x_1, \dots, x_n$

$\Omega(n \log n)$



# CONVEX HULL IN 2D

## Extension: convex hull of a simple polygon

- Is it possible to design an  $o(n \log n)$  time algorithm by exploiting the order of the vertices of the polygon?
- Is it possible, for example, to apply Graham's algorithm using the order of the vertices of the polygon?

# CONVEX HULL IN 2D

## Extension: convex hull of a simple polygon

- Is it possible to design an  $o(n \log n)$  time algorithm by exploiting the order of the vertices of the polygon?
- Is it possible, for example, to apply Graham's algorithm using the order of the vertices of the polygon?

## Extension: do the previous strategies extend to the 3D case?

- Is it possible to design a 3-dimensional gift wrap convex hull algorithm?
- Is it possible to design a 3-dimensional incremental convex hull algorithm?
- Is it possible to design a 3-dimensional divide-and-conquer convex hull algorithm?

# CONVEX HULL IN 2D

## FURTHER READING

- J. O'Rourke, **Computational Geometry in C (2nd ed.)**, Cambridge University Press, 1998.
- F. Preparata, M. Shamos, **Computational Geometry: An introduction (revised ed.)**, Springer, 1993.

## ...AND PLAYING

In 2D:

<http://www.dma.fi.upm.es/docencia/segundociclo/geomcomp/convex.html>

In 3D:

<http://www.cse.unsw.edu.au/~lambert/java/3d>