impa

30º Colóquio Brasileiro de Matemática

IMPA, Rio de Janeiro, 26 a 31 de julho de 2015

Aula 3

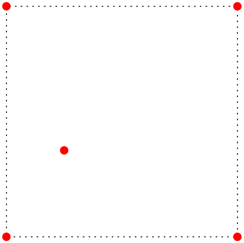
Triangulações

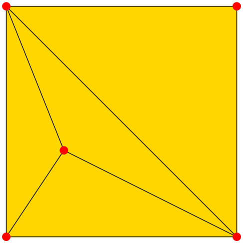
Aula 3 - 28 de julho de 2015 - 12h30min às 14h

Aula 3 – Triangulações

- O problema da triangulação de um conjunto de pontos em $\mathbb{E}^2$

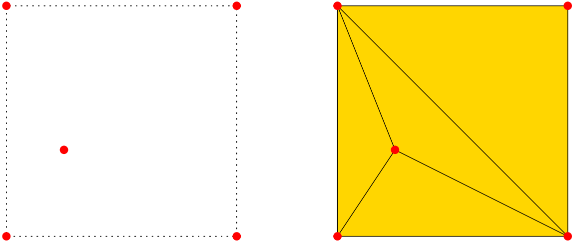
- O problema da triangulação de um conjunto de pontos em $\mathbb{E}^2$
  - Dado um conjunto, P, de pontos em $\mathbb{E}^2$, calcule uma triangulação de P.





Aula 3 – Triangulações

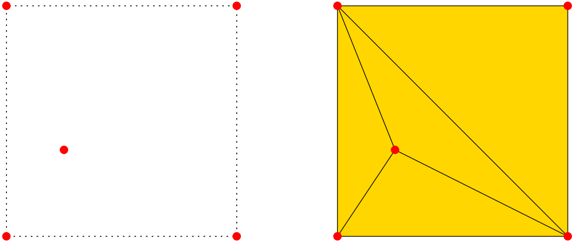
- O problema da triangulação de um conjunto de pontos em $\mathbb{E}^2$
  - Dado um conjunto, P, de pontos em $\mathbb{E}^2$, calcule uma triangulação de P.



- Este problema sempre admite solução?

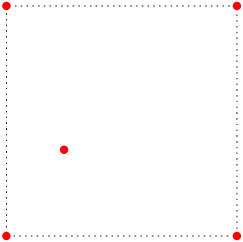
Aula 3 – Triangulações

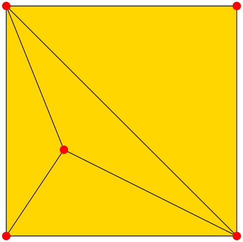
- O problema da triangulação de um conjunto de pontos em $\mathbb{E}^2$
  - Dado um conjunto, P, de pontos em $\mathbb{E}^2$, calcule uma triangulação de P.



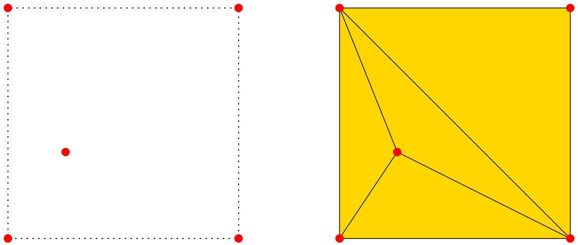
- Este problema sempre admite solução?
- Se sim, a solução é sempre única?

- **Teorema** (existência)
  Todo conjunto finito e não vazio $P \subset \mathbb{E}^2$ admite uma triangulação.





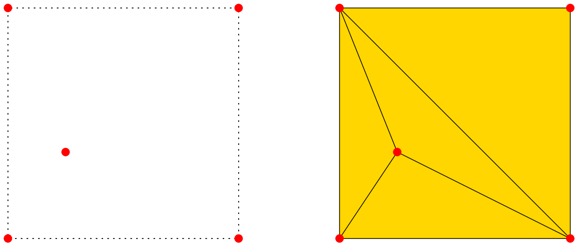
- **Teorema** (existência)
  Todo conjunto finito e não vazio $P \subset \mathbb{E}^2$ admite uma triangulação.



- Em geral, existem muitas triangulações de P.

Aula 3 – Triangulações

- **Teorema** (existência)
  Todo conjunto finito e não vazio $P \subset \mathbb{E}^2$ admite uma triangulação.



- Em geral, existem muitas triangulações de P.
- Veremos uma solução particular: a *triangulação de Delaunay*.

Aula 3 – Triangulações

- Suponha que

(C1) P possui pelo menos quatro pontos

- Suponha que
  (C1) P possui pelo menos quatro pontos
  (C2) Nenhum subconjunto de 4 pontos de P está na mesma circunferência

Aula 3 – Triangulações

- Suponha que

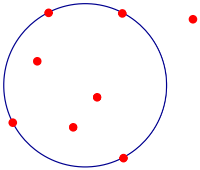
  (C1) P possui pelo menos quatro pontos

  (C2) Nenhum subconjunto de 4 pontos de P está na mesma circunferência

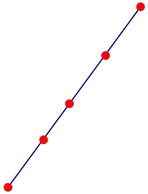
  (C3) Três pontos de P formam um conjunto AI em $\mathbb{E}^2$

Aula 3 – Triangulações

- Suponha que
  - (C1) P possui pelo menos quatro pontos
  - (C2) Nenhum subconjunto de 4 pontos de P está na mesma circunferência
  - (C3) Três pontos de P formam um conjunto AI em $\mathbb{E}^2$

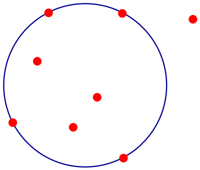


Corcicularidade

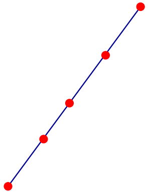


Colinearidade

- Suponha que
  - (C1) P possui pelo menos quatro pontos
  - (C2) Nenhum subconjunto de 4 pontos de P está na mesma circunferência
  - (C3) Três pontos de P formam um conjunto AI em $\mathbb{E}^2$



Corcicularidade



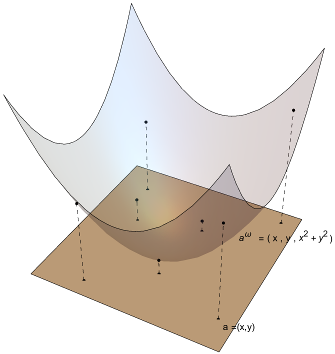
Colinearidade

- Se as condições (C1)-(C3) forem satisfeitas, então podemos definir, de forma única, a triangulação de Delaunay de P, denotada por $\mathcal{TD}(P)$.

Aula 3 – Triangulações

- Seja
$$\omega : \mathbb{E}^2 \to \mathbb{R}$$
tal que
$$\omega(a) = \omega(x, y) = x^2 + y^2 \,,$$
para todo ponto a de $\mathbb{E}^2$ cujas coordenadas cartesianas são (x, y).

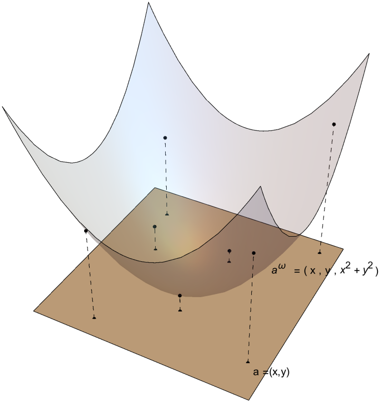
- Seja
$$\omega : \mathbb{E}^2 \to \mathbb{R}$$
tal que
$$\omega(a) = \omega(x, y) = x^2 + y^2 ,$$
para todo ponto a de $\mathbb{E}^2$ cujas coordenadas cartesianas são (x, y).



- Note que

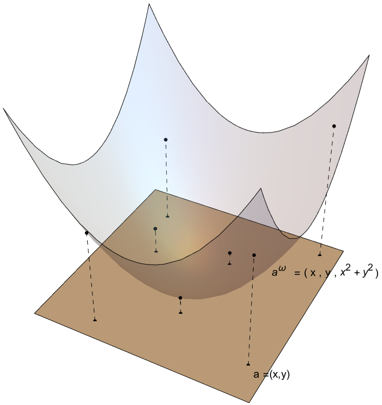
$$a^{\omega} = (x, y, \omega(x, y)) \in \mathbb{E}^3$$

é um ponto do gráfico do parabolóide definido por $x^2 + y^2 = 0$.

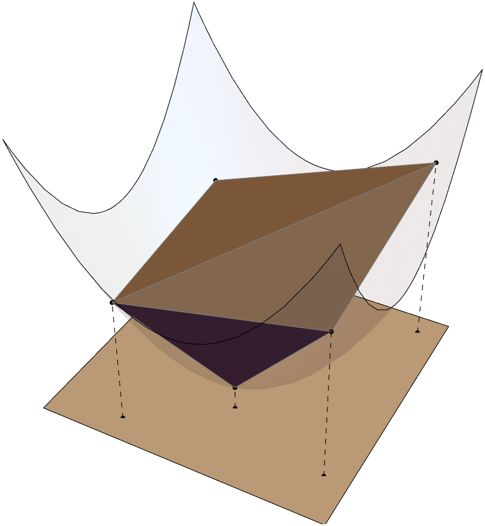


- A função ω é conhecida como mapa de elevação.

- A função ω é conhecida como mapa de elevação.
- Seja
$$P^{\omega} = \{a^{\omega} = (x, y, \omega(x, y)) \mid a = (x, y) \in P\}.$$

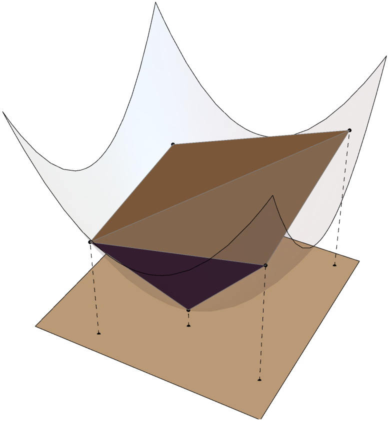


- O fecho convexo, $FC(P^\omega)$, de P^ω é um poliedro em $\mathbb{E}^3$:

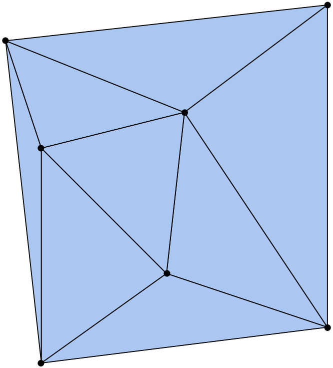


Aula 3 – Triangulações

- A projeção ortogonal do *envelope inferior* de P^ω sobre o plano XY define uma triangulação de P em $\mathbb{E}^2$: a triangulação de Delaunay de P, $\mathcal{TD}(P)$.



- A projeção ortogonal do *envelope inferior* de P^ω sobre o plano XY define uma triangulação de P em $\mathbb{E}^2$: a triangulação de Delaunay de P, $\mathcal{TD}(P)$.



Aula 3 – Triangulações

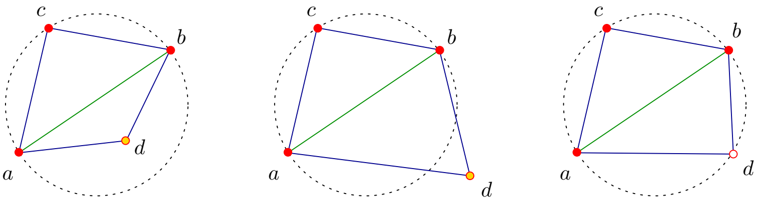
- O que há de especial sobre $\mathcal{TD}(P)$?

Aula 3 – Triangulações

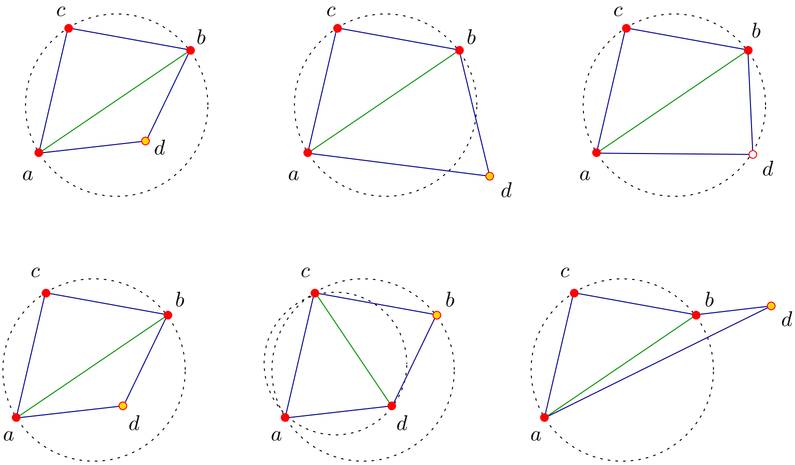
- O que há de especial sobre $\mathcal{TD}(P)$?
- Toda aresta de $\mathcal{TD}(P)$ é localmente Delaunay.

Aula 3 – Triangulações

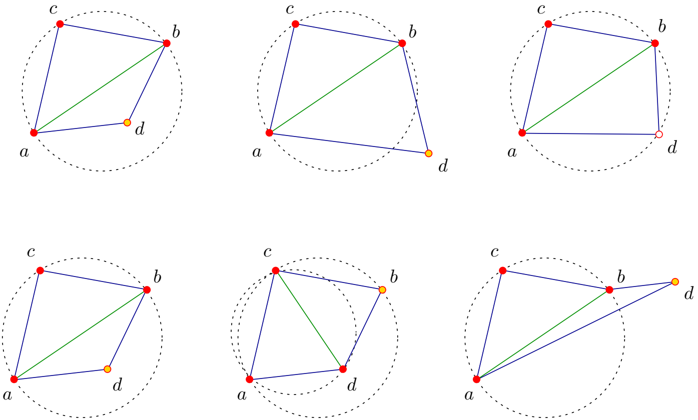
- O que há de especial sobre $\mathcal{TD}(P)$?
- Toda aresta de $\mathcal{TD}(P)$ é localmente Delaunay.



- O que há de especial sobre $\mathcal{TD}(P)$?
- Toda aresta de $\mathcal{TD}(P)$ é localmente Delaunay.



- Por outro lado, seja $\mathcal{T}(P)$ qualquer triangulação de P. Se toda aresta de $\mathcal{T}(P)$ é localmente Delaunay, então podemos mostrar que $\mathcal{T}(P)$ é $\mathcal{TD}(P)$.



- Dado P satisfazendo (C1)-(C3), como podemos construir $\mathcal{TD}(P)$?

Aula 3 – Triangulações

- Dado P satisfazendo (C1)-(C3), como podemos construir $\mathcal{TD}(P)$?
- Existem diversos "algoritmos" para construir $\mathcal{TD}(P)$, mas um deles, denominado algoritmo (de inserção) incremental, é o mais apropriado para atingir o nosso principal objetivo: *construir (ou gerar) malhas em* $\mathbb{E}^2$.

- Dado P satisfazendo (C1)-(C3), como podemos construir $\mathcal{TD}(P)$?
- Existem diversos "algoritmos" para construir $\mathcal{TD}(P)$, mas um deles, denominado algoritmo (de inserção) incremental, é o mais apropriado para atingir o nosso principal objetivo: *construir (ou gerar) malhas em* $\mathbb{E}^2$.
- Seja

$$p_{-2}, p_{-1}, p_0, p_1, \ldots, p_n$$

uma sequência formada pelos pontos $p_1, \ldots, p_n$ de P e pelos pontos

$$p_{-2} = (-M, -M), \quad p_{-1} = (0, M) \quad \text{e} \quad p_0 = (M, 0)$$

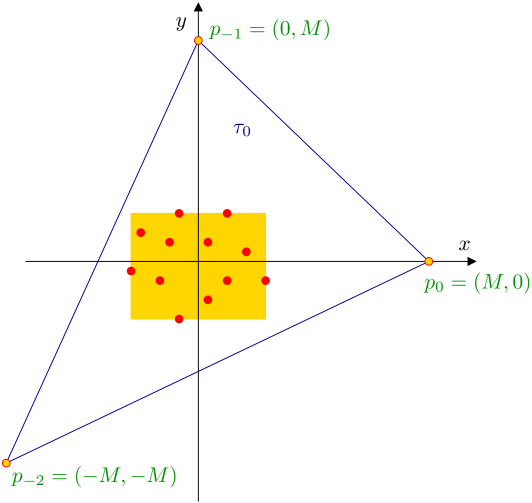
tais que

$$M = 3 \cdot \max_{i=1}^{n} \{|x_i|, |y_i|\},$$

em que (x_i, y_i) são as coordenadas cartesianas do ponto p_i de P, para todo

$$i = 1, \ldots, n.$$

Aula 3 – Triangulações



Aula 3 – Triangulações

- Os pontos p_{-2}, p_{-1} e p_0 são denominados pontos especiais.

- Os pontos p_{-2}, p_{-1} e p_0 são denominados pontos especiais.
- O algoritmo incremental constrói uma sequência de triangulações,

$$\mathcal{TD}_0, \mathcal{TD}_1, \ldots, \mathcal{TD}_n\,,$$

tal que

$$\mathcal{TD}_0 = \mathcal{TD}(P_0) = \mathcal{TD}(\{p_{-2}, p_{-1}, p_0\})$$

e

$$\mathcal{TD}_j = \mathcal{TD}(P_j) = \mathcal{TD}(P_{j-1} \cup \{p_j\})\,, \quad \forall j = 1, \ldots, n\,.$$

Aula 3 – Triangulações

- Os pontos p_{-2}, p_{-1} e p_0 são denominados pontos especiais.
- O algoritmo incremental constrói uma sequência de triangulações,

$$\mathcal{TD}_0, \mathcal{TD}_1, \ldots, \mathcal{TD}_n\,,$$

  tal que

$$\mathcal{TD}_0 = \mathcal{TD}(P_0) = \mathcal{TD}(\{p_{-2}, p_{-1}, p_0\})$$

  e

$$\mathcal{TD}_j = \mathcal{TD}(P_j) = \mathcal{TD}(P_{j-1} \cup \{p_j\})\,, \quad \forall j = 1, \ldots, n\,.$$

- Observe que

$$\mathcal{TD}_n = \mathcal{TD}(P_n) = \mathcal{TD}(P \cup \{p_{-2}, p_{-1}, p_0\})\,.$$

Aula 3 – Triangulações

- Os pontos p_{-2}, p_{-1} e p_0 são denominados pontos especiais.
- O algoritmo incremental constrói uma sequência de triangulações,

$$\mathcal{TD}_0, \mathcal{TD}_1, \ldots, \mathcal{TD}_n\,,$$

tal que

$$\mathcal{TD}_0 = \mathcal{TD}(P_0) = \mathcal{TD}(\{p_{-2}, p_{-1}, p_0\})$$

e

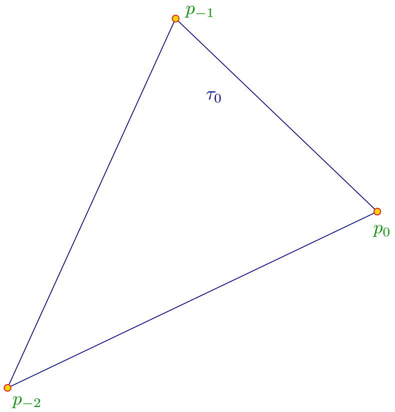
$$\mathcal{TD}_j = \mathcal{TD}(P_j) = \mathcal{TD}(P_{j-1} \cup \{p_j\})\,, \quad \forall j = 1, \ldots, n\,.$$

- Observe que

$$\mathcal{TD}_n = \mathcal{TD}(P_n) = \mathcal{TD}(P \cup \{p_{-2}, p_{-1}, p_0\})\,.$$

- A triangulação $\mathcal{TD}_j$ é construída a partir da triangulação $\mathcal{TD}_{j-1}$ e p_{j-1}.

- A triangulação $\mathcal{TD}_0$ contém apenas um triângulo: $\tau_0 = [p_{-2}, p_{-1}, p_0]$.



- Vamos agora proceder de forma indutiva.

Aula 3 – Triangulações

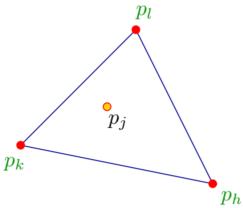
- Vamos agora proceder de forma indutiva.
- Suponha que $\mathcal{TD}_{j-1} = \mathcal{TD}(P_{j-1})$ tenha sido construída.

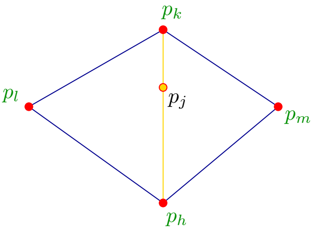
- Vamos agora proceder de forma indutiva.
- Suponha que $\mathcal{TD}_{j-1} = \mathcal{TD}(P_{j-1})$ tenha sido construída.
- Queremos construir $\mathcal{TD}_j = \mathcal{TD}(P_{j-1} \cup \{p_j\}) = \mathcal{TD}(P_j)$ a partir de $\mathcal{TD}_{j-1}$.

Aula 3 – Triangulações

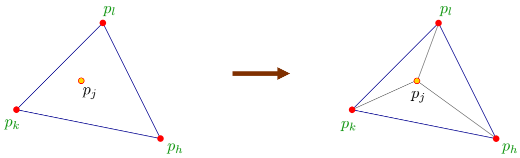
- Vamos agora proceder de forma indutiva.
- Suponha que $\mathcal{TD}_{j-1} = \mathcal{TD}(P_{j-1})$ tenha sido construída.
- Queremos construir $\mathcal{TD}_j = \mathcal{TD}(P_{j-1} \cup \{p_j\}) = \mathcal{TD}(P_j)$ a partir de $\mathcal{TD}_{j-1}$.
- O primeiro passo é localizar um triângulo de $\mathcal{TD}_{j-1}$ contendo p_j.

- Vamos agora proceder de forma indutiva.
- Suponha que $\mathcal{TD}_{j-1} = \mathcal{TD}(P_{j-1})$ tenha sido construída.
- Queremos construir $\mathcal{TD}_j = \mathcal{TD}(P_{j-1} \cup \{p_j\}) = \mathcal{TD}(P_j)$ a partir de $\mathcal{TD}_{j-1}$.
- O primeiro passo é localizar um triângulo de $\mathcal{TD}_{j-1}$ contendo p_j.
- Há dois possíveis cenários:

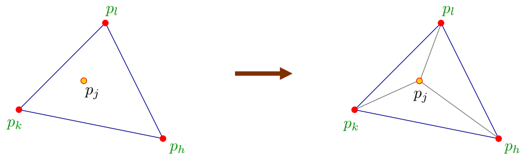




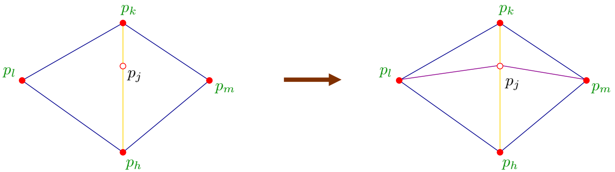
- No primeiro cenário, dividimos o triângulo em 3:



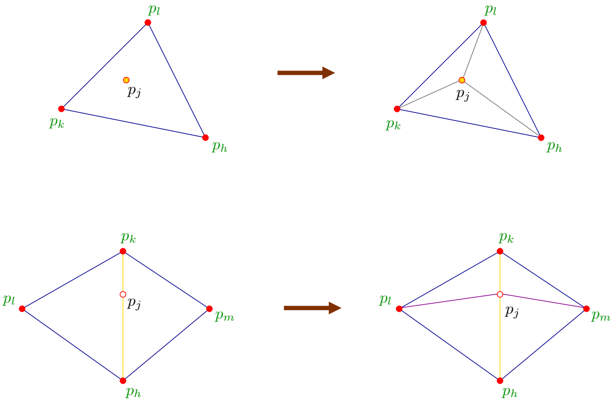
- No primeiro cenário, dividimos o triângulo em 3:



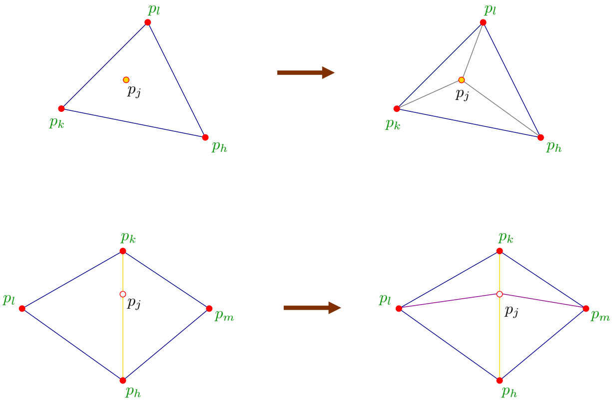
- No segundo, dividimos os 2 triângulos em 2 novos triângulos cada:



- Em ambos, a triangulação resultante da *inserção do vértice* p_j em $\mathcal{TD}_{j-1}$ **não é** necessariamente a triangulação de Delaunay, $\mathcal{TD}(P_j)$, de P_j.



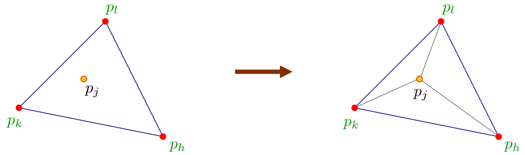
- Mas, nós podemos transformar a triangulação resultante em uma triangulação de Delaunay, *a triangulação* $\mathcal{TD}_j$, através da operação trocas de arestas.

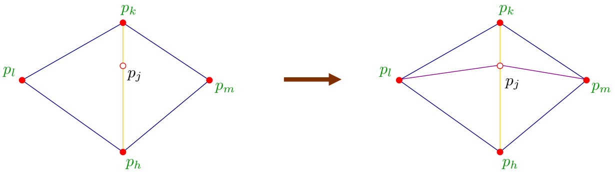


Aula 3 – Triangulações

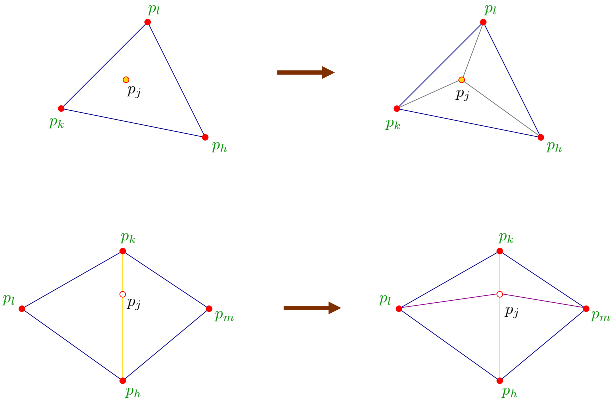
- O que isso significa?

- O que isso significa?
- Tornar todas as arestas localmente Delaunay!





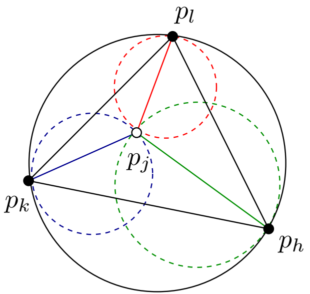
- Para tal, precisamos identificar e substituir as arestas da triangulação resultante que **não são** localmente Delaunay. Como podemos fazer isso?

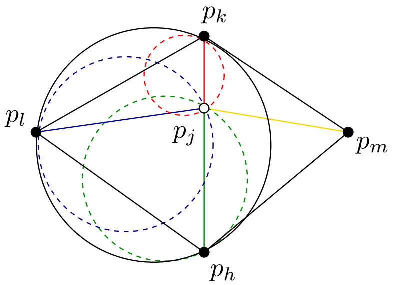


- Primeiro, notamos que as arestas criadas durante a inserção de p_j são, todas elas, localmente Delaunay. Por que esta afirmação é verdadeira?

Aula 3 – Triangulações

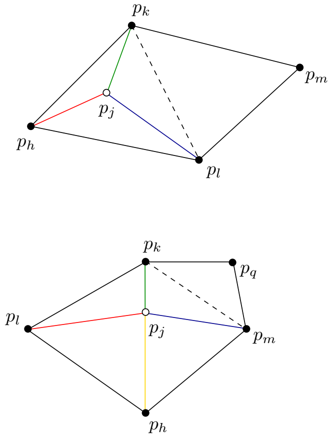
- Primeiro, notamos que as arestas criadas durante a inserção de p_j são, todas elas, localmente Delaunay. Por que esta afirmação é verdadeira?



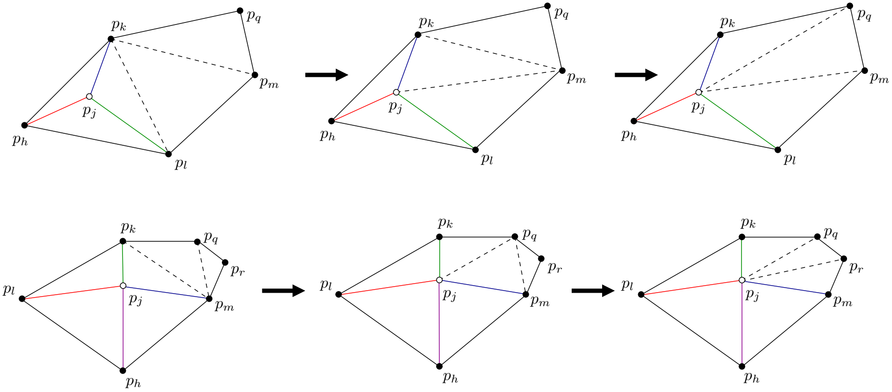


- Quais arestas podem não ser localmente Delaunay?

- Quais arestas podem não ser localmente Delaunay?



- Quais arestas podem não ser localmente Delaunay?

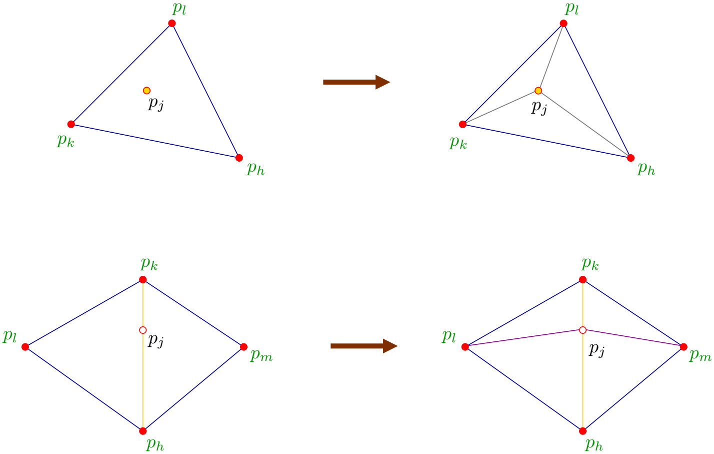


Aula 3 – Triangulações

Algoritmo 1 $\textsc{InserePonto}(p, \mathcal{T})$

encontre um triângulo $\tau = [p_h, p_k, p_l]$ de $\mathcal{T}$ contendo p
se p pertence ao interior de τ **então**
    insira p em $\mathcal{T}$ e gere $\mathcal{T}'$
    $\textsc{TestaTroca}(p, p_h, p_k, \mathcal{T}')$
    $\textsc{TestaTroca}(p, p_k, p_l, \mathcal{T}')$
    $\textsc{TestaTroca}(p, p_l, p_h, \mathcal{T}')$
senão
    seja $\nu = [p_h, p_k]$ a aresta de τ contendo p
    seja $\sigma = [p_h, p_k, p_m]$ o outro triângulo de $\mathcal{T}$ contendo ν
    insira p em $\mathcal{T}$ e gere $\mathcal{T}'$
    $\textsc{TestaTroca}(p_j, p_h, p_l, \mathcal{T}')$
    $\textsc{TestaTroca}(p_j, p_l, p_k, \mathcal{T}')$
    $\textsc{TestaTroca}(p_j, p_k, p_m, \mathcal{T}')$
    $\textsc{TestaTroca}(p_j, p_m, p_h, \mathcal{T}')$
fim se
devolva $\mathcal{T}'$

Aula 3 – Triangulações



Aula 3 – Triangulações

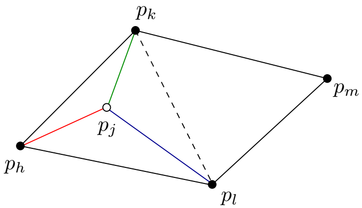
Algoritmo 1 $\textsc{InserePonto}(p, \mathcal{T})$

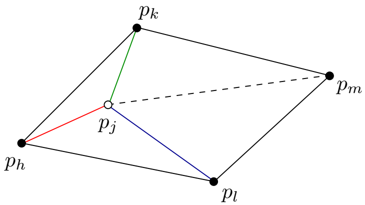
encontre um triângulo $\tau = [p_h, p_k, p_l]$ de $\mathcal{T}$ contendo p
se p pertence ao interior de τ **então**
    insira p em $\mathcal{T}$ e gere $\mathcal{T}'$
    $\textsc{TestaTroca}(p, p_h, p_k, \mathcal{T}')$
    $\textsc{TestaTroca}(p, p_k, p_l, \mathcal{T}')$
    $\textsc{TestaTroca}(p, p_l, p_h, \mathcal{T}')$
senão
    seja $\nu = [p_h, p_k]$ a aresta de τ contendo p
    seja $\sigma = [p_h, p_k, p_m]$ o outro triângulo de $\mathcal{T}$ contendo ν
    insira p em $\mathcal{T}$ e gere $\mathcal{T}'$
    $\textsc{TestaTroca}(p_j, p_h, p_l, \mathcal{T}')$
    $\textsc{TestaTroca}(p_j, p_l, p_k, \mathcal{T}')$
    $\textsc{TestaTroca}(p_j, p_k, p_m, \mathcal{T}')$
    $\textsc{TestaTroca}(p_j, p_m, p_h, \mathcal{T}')$
fim se
devolva $\mathcal{T}'$

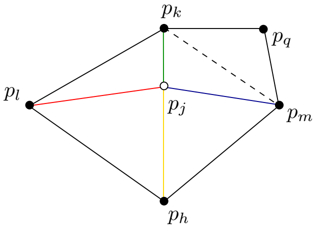
Algoritmo 2 $\textsc{TrocaArestas}(p, a, b, \mathcal{T})$

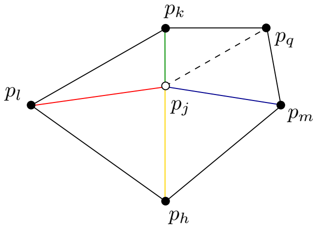
se $[a, b]$ é uma aresta da fronteira de $\mathcal{T}$ **então**
    devolva
fim se
seja $[a, c, b]$ o triângulo que compartilha $[a, b]$ com $[a, b, p]$
se $\textsc{InCircle}(a, c, b, p)$ **então**
    troque a aresta $[a, b]$ pela aresta $[p, c]$ em $\mathcal{T}$
    $\textsc{TrocaArestas}(p, a, c, \mathcal{T})$
    $\textsc{TrocaArestas}(p, c, b, \mathcal{T})$
fim se
devolva

Aula 3 – Triangulações

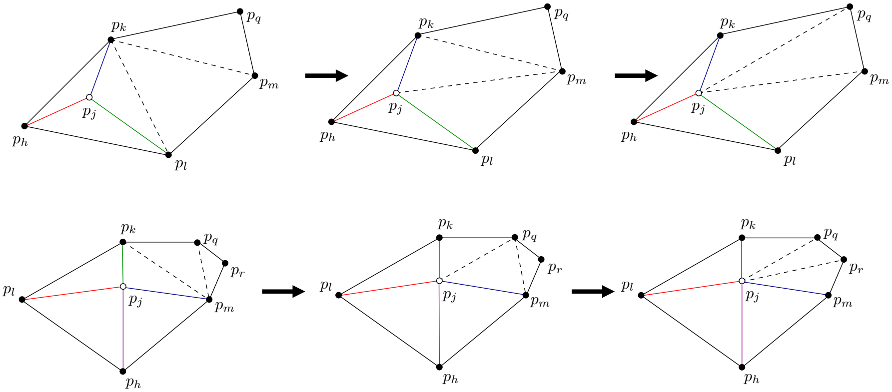








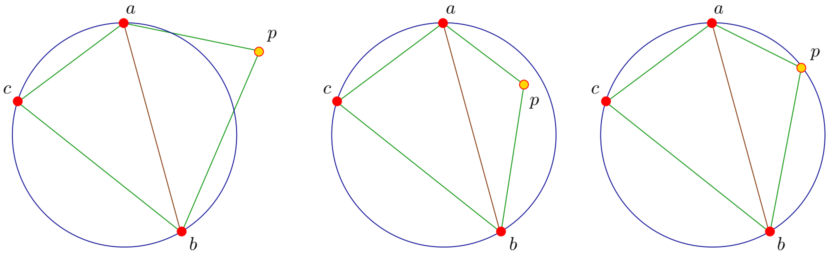
Aula 3 – Triangulações



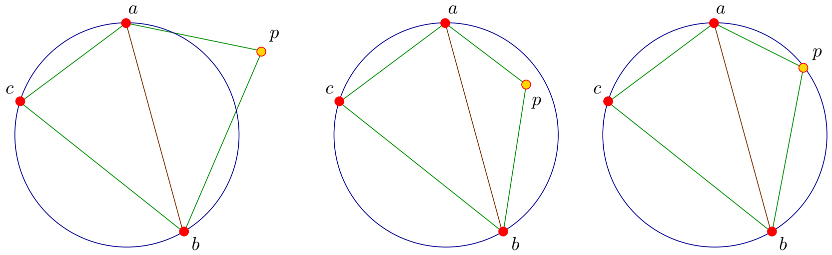
Algoritmo 2 TrocaArestas($p, a, b, \mathcal{T}$)

```
se [a, b] é uma aresta da fronteira de T então
    devolva
fim se
seja [a, c, b] o triângulo que compartilha [a, b] com [a, b, p]
se InCircle(a, c, b, p) então
    troque a aresta [a, b] pela aresta [p, c] em T
    TrocaArestas(p, a, c, T)
    TrocaArestas(p, c, b, T)
fim se
devolva
```

- O teste InCircle(a, c, b, p)

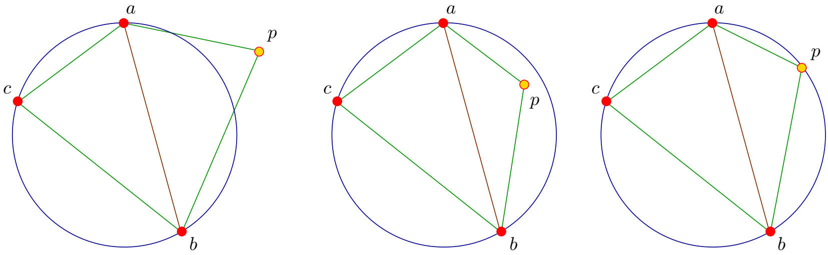


- O teste $\textsc{InCircle}(a, c, b, p)$



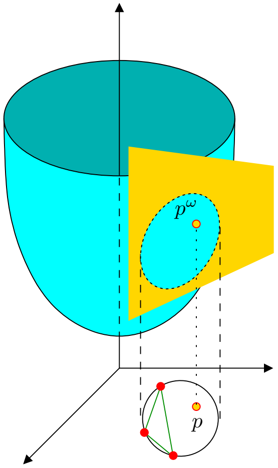
- Como efetuamos este teste?

- O teste InCircle(a, c, b, p)



- Como efetuamos este teste?
- Há várias maneiras e a escolha é influenciada pela robustez numérica.

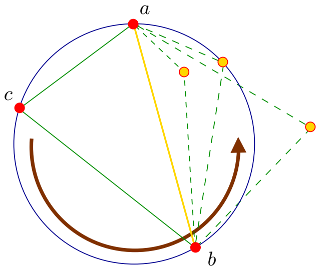
- O teste $\textsc{InCircle}(a, c, b, p)$



Aula 3 – Triangulações

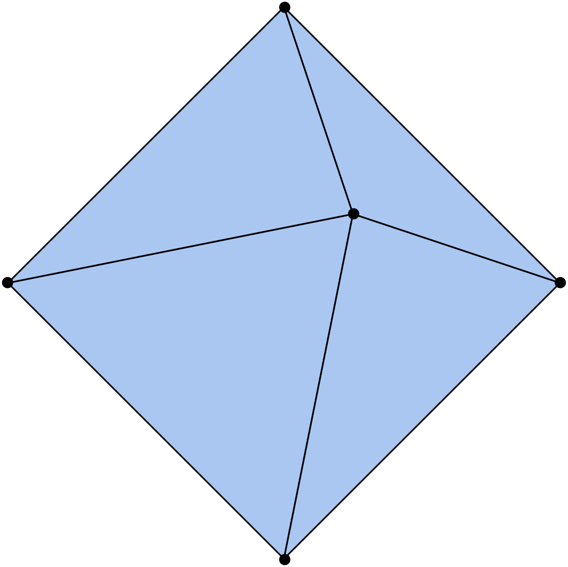
- O teste $\textsc{InCircle}(a, c, b, p)$

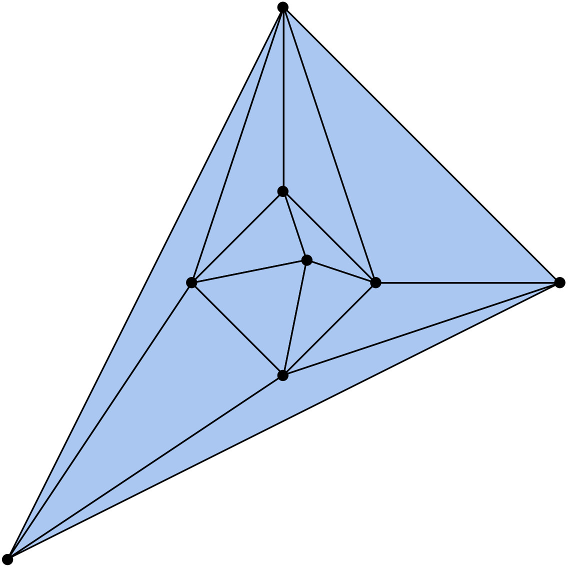
$$\textsc{InCircle}(a, c, b, p) = \begin{vmatrix} x_a & y_a & x_a^2 + y_a^2 & 1 \\ x_c & y_c & x_c^2 + y_c^2 & 1 \\ x_b & y_b & x_b^2 + y_b^2 & 1 \\ x_p & y_p & x_p^2 + y_p^2 & 1 \end{vmatrix} <, > \text{ ou } = 0?$$

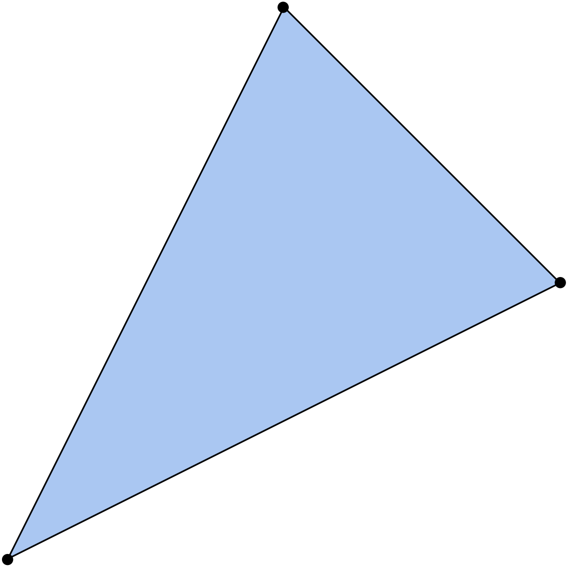


Algoritmo 3 $\textsc{CalculaTriDel}(P)$

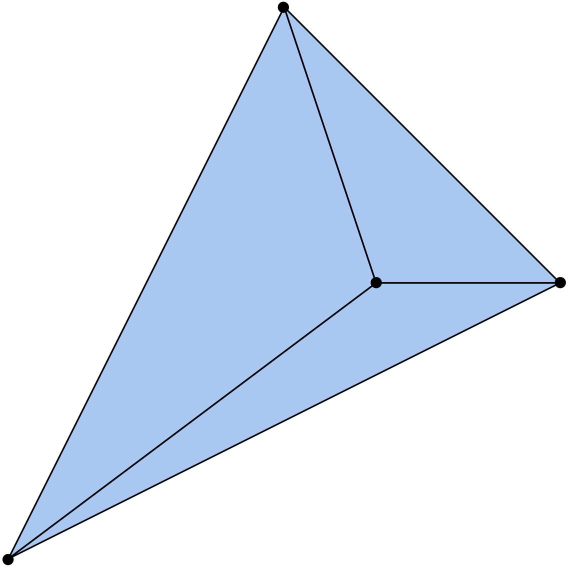
gere uma sequência $p_1, \ldots, p_n$ a partir de uma permutação de P
calcule as coordenadas dos pontos especiais p_{-2}, p_{-1} e p_0
gere a triangulação $\mathcal{TD}_0$
para $j = 1, \ldots, |P|$ **faça**
    $\mathcal{TD}_j \leftarrow \textsc{InserePonto}(p_j, \mathcal{TD}_{j-1})$
fim para
remova p_{-2}, p_{-1} e p_0 de $\mathcal{TD}_{|P|}$
devolva $\mathcal{TD}_{|P|}$



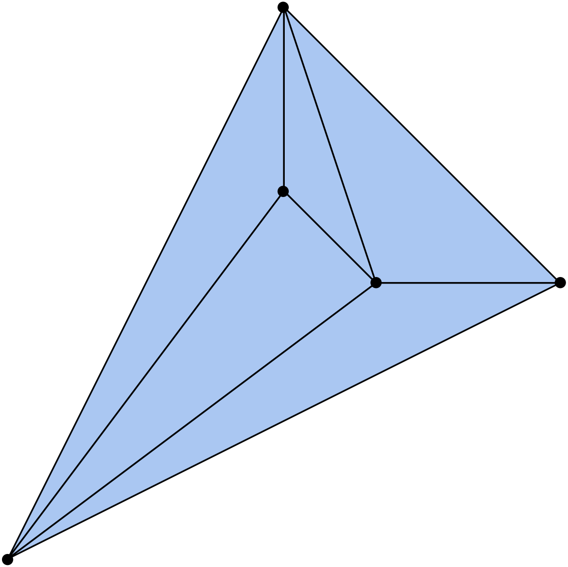


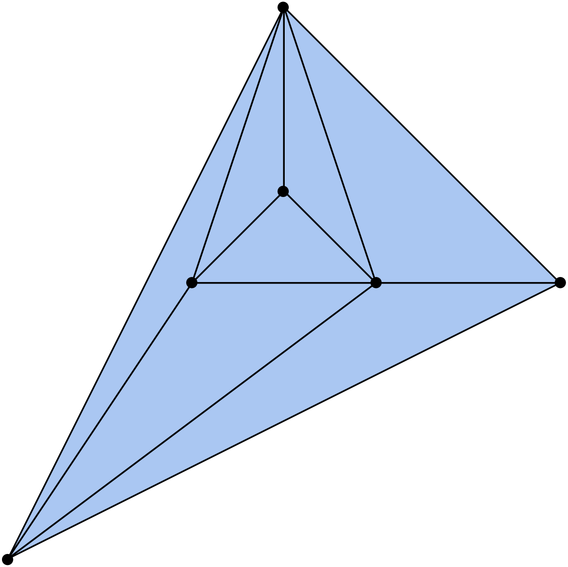


Aula 3 – Triangulações

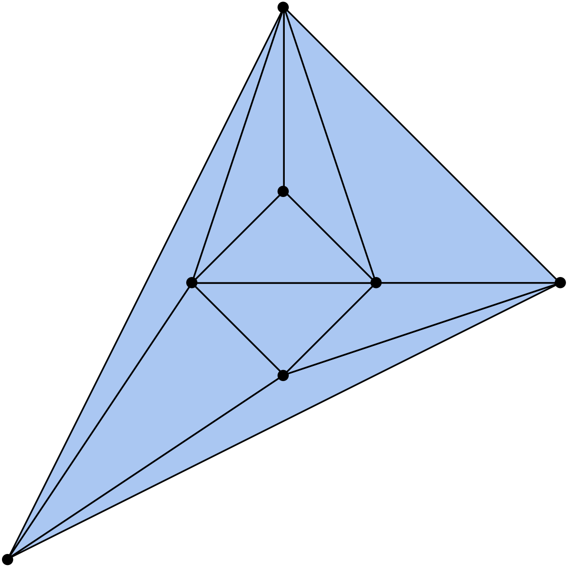


Aula 3 – Triangulações

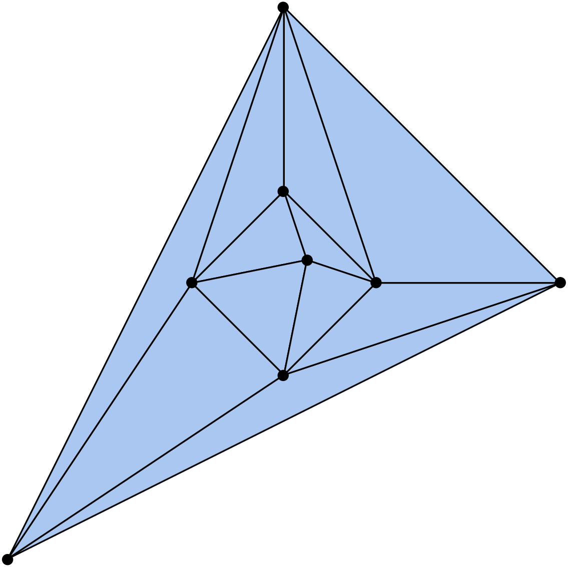




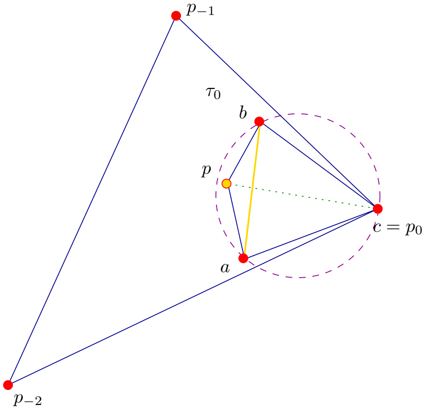
Aula 3 – Triangulações

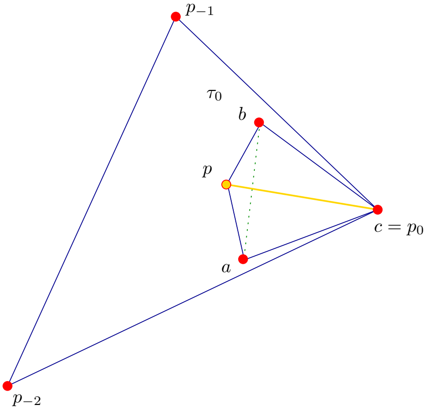


Aula 3 – Triangulações

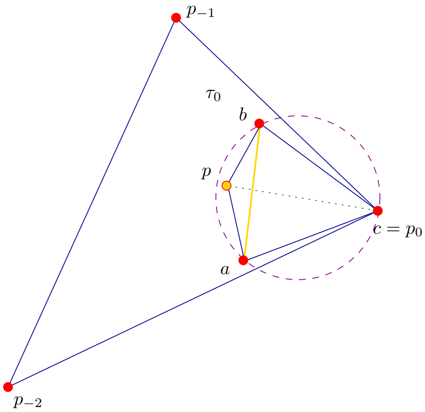


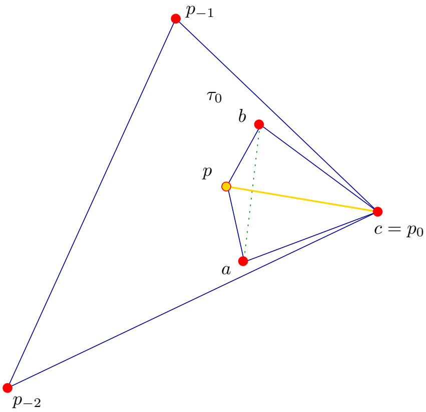
- A remoção dos pontos especiais não é tão simples assim...



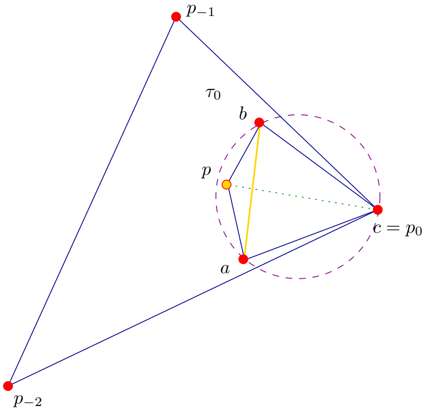


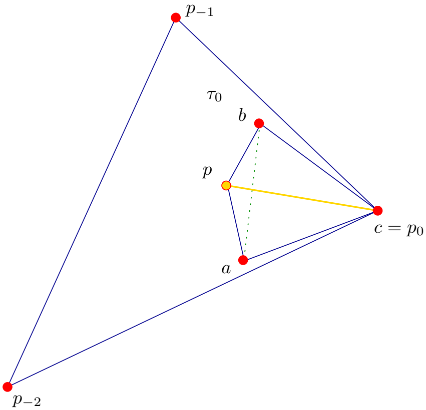
- O segredo é tratar os pontos especiais de forma "simbólica"...





- Podemos fazer isso modificando o predicado INCIRCLE().





Aula 3 – Triangulações

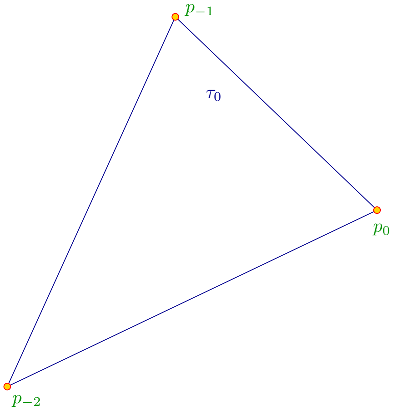
Algoritmo 4 InCircleMod(a, c, b, p)

se nenhum de a, c e b é especial **então**
    devolva InCircle(a, c, b, p)
fim se
se exatamente um de a, b e c é especial **então**
    se c é especial **então**
        devolva **falso**{apenas c é especial}
    senão
        {exatamente um de a e b é especial e c não é}
        $d1 \leftarrow$ Left(c, b, p)
        $d2 \leftarrow$ LeftOn(c, a, p)
        {verdadeiro se, e somente se, $\square(a, c, b, p)$ é convexo}
        devolva $d1$ **and** (**not** $d2$)
    fim se
fim se
devolva $in(c) < \min\{in(a), in(b)\}$ {a e c ou b e c são especiais}

- Algumas perguntas precisam ser respondidas:

Aula 3 – Triangulações

- Algumas perguntas precisam ser respondidas:
  - Por que utilizar pontos especiais?

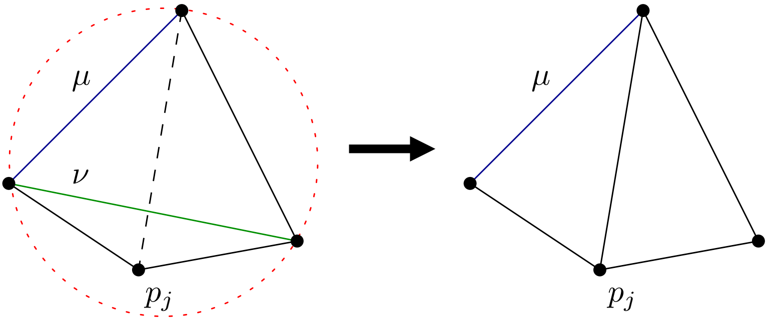


Aula 3 – Triangulações

- Algumas perguntas precisam ser respondidas:

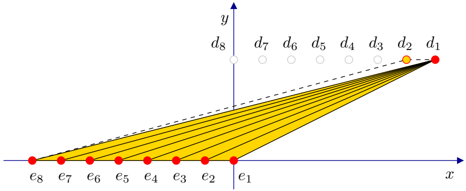
- Algumas perguntas precisam ser respondidas:
    - Por que a triangulação produzida por $\textsc{InserePonto}(p, \mathcal{T})$ é Delaunay?

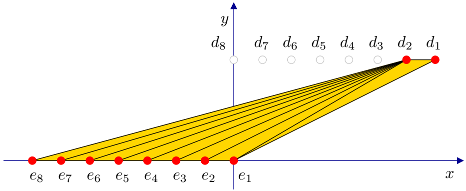
- Algumas perguntas precisam ser respondidas:
  - Por que a triangulação produzida por $\textsc{InserePonto}(p, \mathcal{T})$ é Delaunay?



- Algumas perguntas precisam ser respondidas:
    - Quão eficiente é o algoritmo $\textsc{CalculaTriDel}(P)$?

- Algumas perguntas precisam ser respondidas:
    - Quão eficiente é o algoritmo CALCULATRIDEL(P)?





Exercícios

- Sugerimos a resolução dos seguintes problemas do livro:
  - 4.3
  - 4.4
  - 4.13
  - 4.15
  - 4.27
  - 4.38
  - 4.39
  - 4.41
  - 4.42

Exercícios

- Sugerimos a resolução dos seguintes problemas do livro:
  - 4.3
  - 4.4
  - 4.13
  - 4.15
  - 4.27
  - 4.38
  - 4.39
  - 4.41
  - 4.42
- Os problemas acima estão relacionados ao assunto da aula 3.