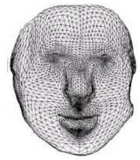
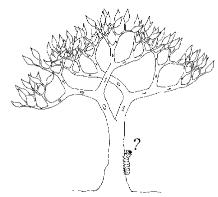


2. Alguns Problemas Elementares da Geometria Computacional

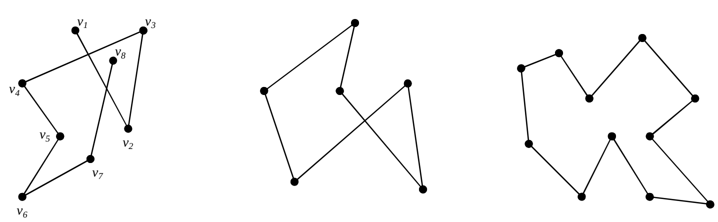
Antonio Leslie Bajuelos
Departamento de Matemática
Universidade de Aveiro



Mestrado em Matemática e Aplicações

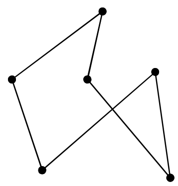
Preliminares

- Uma **curva poligonal** (ou **cadeia poligonal**) é uma sequência finita $v_0, e_0, v_1, \ldots, e_{n-1}, v_n$ onde
  - $v_0, \ldots, v_n$ são pontos de $\Re^2$ e
  - e_i é um segmento de recta com extremidades v_i e v_{i+1} ($i = 0, \ldots, n-1$), de forma que, os segmentos de recta tenham dois a dois, um extremo em comum e não existam dois segmentos com o mesmo extremo na mesma recta, ou seja, tenham três pontos não colineares.
- Os pontos $v_0, \ldots, v_n$ também são designados **vértices** e os segmentos de recta $e_0, \ldots, e_{n-1}$, **arestas**

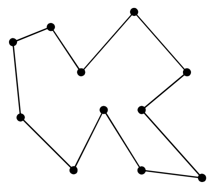


Preliminares

- Uma **curva poligonal** é **fechada** se o último ponto da sequência é igual ao primeiro, ou seja, $v_0 = v_n$.
- Uma **curva poligonal** diz-se **simples** se dados dois segmentos não consecutivos, estes não se intersectam, isto significa que o segmento e_i só intersecta (possivelmente) o segmento e_{i+1} no ponto v_{i+1} ($i = 0, \ldots, n\text{-}1$).



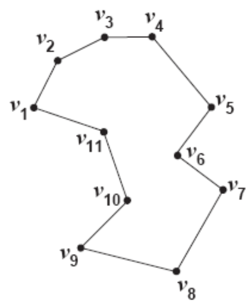
Fechada não simples

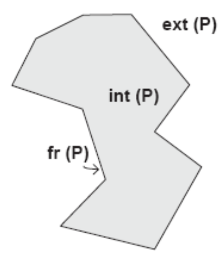


Fechada simples

Preliminares

- **Teorema de Jordan:** Toda a curva plana fechada simples, divide o plano em duas regiões (o **interior** e o **exterior** da curva)
- **Polígono simples**: conjunto dos pontos da **região interior** reunidos com os pontos da **cadeia poligonal simples fechada**

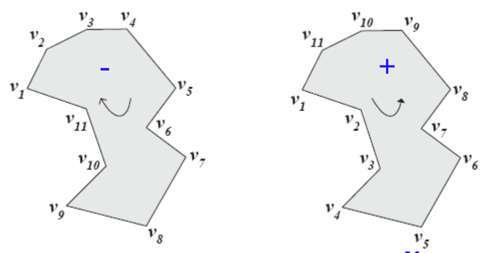




- Denotamos o interior de P por **int(P)**, o exterior de P por **ext(P)** e a fronteira de P por **fr(P)**.

Preliminares

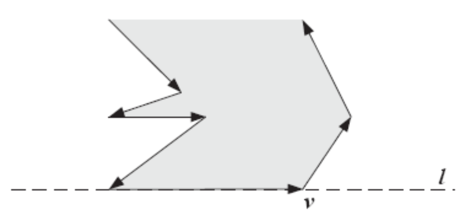
- Um **polígono simples** P fica perfeitamente delimitado por:
  - o conjunto de pontos formado pelos vértices ordenados, quando se percorre a fronteira de P e
  - uma orientação que nos indica onde se irá situar o int(P)
- Orientação **negativa** (**horária** ou **CW**) – o int(P) P fica à direita de qualquer aresta
- Orientação **positiva** (**anti-horária** ou **CCW**) – o int(P) fica à esquerda de qualquer aresta



Por norma suporemos sempre que o polígono simples está orientado positivamente

Preliminares

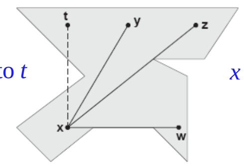
- Um vértice de um polígono diz-se **convexo** se a amplitude do ângulo, pertencente ao seu interior, formado por duas arestas que lhe são incidentes for menor ou igual a π. Caso contrário o vértice diz-se **côncavo** ou **reflexo**
- **Lema:** (Meister) *Qualquer polígono P, tem pelo menos um vértice estritamente convexo.*
  - **Prova:** Seja P um polígono simples. Se percorrermos a fronteira de P, no sentido positivo, quando encontramos um vértice estritamente convexo, temos que virar à esquerda e num vértice estritamente côncavo viramos à direita. Dos vértices de P com ordenada (y) mínima seja v o que tem abcissa (x) máxima. Seja l a recta horizontal passando sobre v. A aresta seguinte a v, que lhe está incidente, está acima de l (ver figura). Logo, deveremos virar à esquerda em v. Portanto, v é um vértice estritamente convexo. ♦



Preliminares

- **Visibilidade**: dois pontos x e y, num polígono simples P são visíveis se o segmento de recta $xy \subset$ P

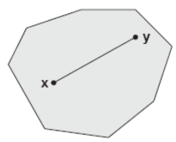
x não vê o ponto *t*

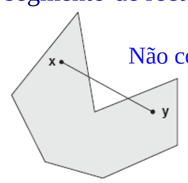


x vê *y*, *z* e *w*

- **Polígono convexo**: se para todo $x, y \in$ P, o segmento de recta $xy \subset$ P.

Convexo





Não convexo

- **Teorema:** *Um polígono é convexo sse não tem vértices côncavos*

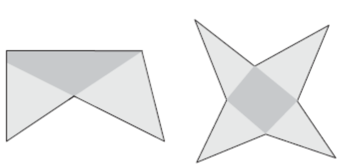
Preliminares

- **Polígono estrelado:**
  - Se existe pelo menos um ponto $x \in$ P tal que para todo ponto $y \in$ P o segmento $xy \subset$ P

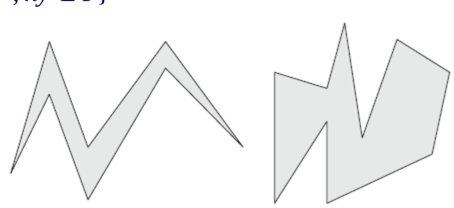
  ou

  - Se existir pelo menos um ponto do qual é possível ver todo o polígono

- **Núcleo:** Ker(P) = $\{x \in \text{P} \mid \forall\, y \in \text{P}, xy \subset \text{P}\}$



Polígonos estrelados e os seus núcleos



Polígonos não estrelados

Preliminares

- **Teorema:** o Ker(P) é um conjunto convexo
  - **Prova:** pela definição de Ker(P)

- **Teorema:** Um polígono P é convexo sse Ker(P) = P
  - **Prova:** Exercício

- **Teorema:** (Krasnosel'ski) Um conjunto S de $\Re^2$ é estrelado sse

  $\forall\, x, y, z \in$ S, $\exists\, w \in$ S tal que *w* vê *x*, *y* e *z*

- **Teorema:** Um polígono P é estrelado sse qualquer terno de vértices convexos é visto por pelo menos um ponto de P
  - **Prova:** deduz-se facilmente da prova do teorema de Krasnosel'ski

Problema Nº 1: Ponto em Polígono

- **Enunciado:** *Dado um polígono simples P no plano e um ponto q do plano, localizar q em relação a P, isto é, decidir se q está no interior de P, no exterior de P, ou na fronteira de P.*
- Duas aplicações:
  - **Sistemas de Informação Geográfica** (SIG ou GIS)
  - **Computação Gráfica** - cada vez que utilizamos o rato para seleccionar um objecto na ecrã do computador.

Problema Nº 1: Ponto em Polígono

- **Parece um problema muito difícil de resolver!!!**

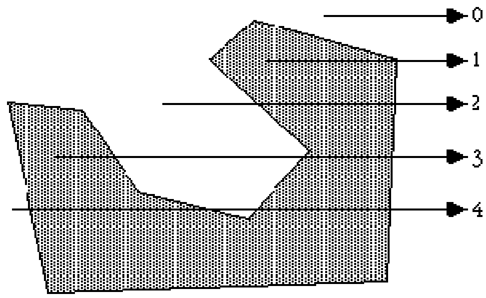
Problema Nº 1: Ponto em Polígono

- **Mão não assim é tão difícil ….**
  - **Princípio fundamental:**
    - Determinar se um ponto $Q(x_q,y_q)$ está ou não dentro de um polígono P, consiste em traçar, a partir de Q, uma **semi-recta** numa direcção qualquer.
    - Esta **semi-recta** estará **alternando** dentro e fora do polígono em questão, mudando de estado nos pontos de intersecção com a fronteira de P **(**fr(P)**)**

M Shimrat, "Algorithm 112, Position of Point Relative to Polygon“, Comm. ACM 5(8), Aug 1962, p 434.

Problema Nº 1: Ponto em Polígono

- **Resumindo as ideias do algoritmo PinP**
  - Se o **número de intersecções** for **ímpar**, o ponto está **dentro (in)** do polígono P
  - Se o **número de intersecções** for **par**, o ponto está **fora (out)** de P



Problema Nº 1: Ponto em Polígono

Par	**→ OUT**
Impar	**→ IN**

- **Esboço do algoritmo PinP**

```
int crossings = 0
  for (each edge of P)
      if (ray down from (x,y) crosses edge)
   crossings ++;
  if (crossings is odd)
          return (inside);
  else
          return (outside);
```

- **Complexidade :** $O(n)$
- **Para mais informação consultar a página web:** http://softsurfer.com/Archive/algorithm_0103/algorithm_0103.htm

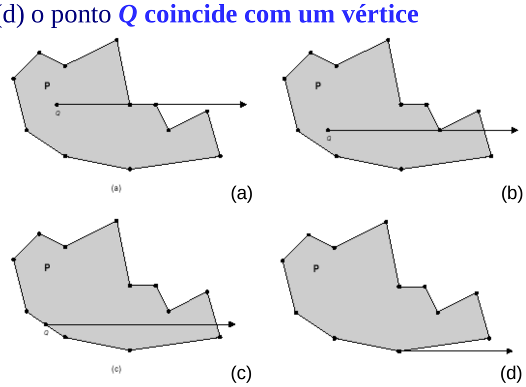
Problema Nº 1: Ponto em Polígono

- Por conveniência, adopta-se em geral **uma semi-recta paralela ao eixo dos *x***, passando por Q e se estendendo para a direita.
- A contagem do **número de intersecções** pode ser feita com facilidade, verificando quantas **arestas** (com pelo menos um extremo à direita de Q ($x \geq x_q$)) de P têm um ponto extremo **acima** e outro **abaixo** da coordenada y_q do ponto Q.

Par	**→ OUT**
Impar	**→ IN**

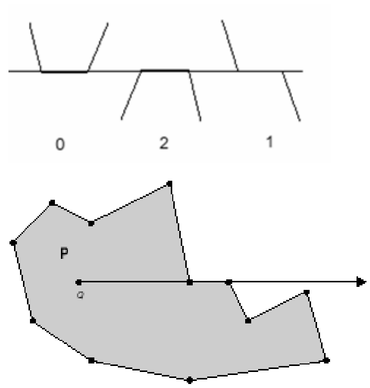
Problema Nº 1: Ponto em Polígono

- A maior dificuldade está no tratamento de **casos degenerados** como:
  - (a) a semi-recta **passa por uma aresta** do polígono
  - (b) a semi-recta **passa por um vértice**
  - (c) o ponto ***Q* está sobre fr(P)**
  - (d) o ponto ***Q* coincide com um vértice**



Problema Nº 1: Ponto em Polígono

(a) Se a semi-recta **passa por uma aresta de P**, então o número de intersecções fica definido da seguinte forma:

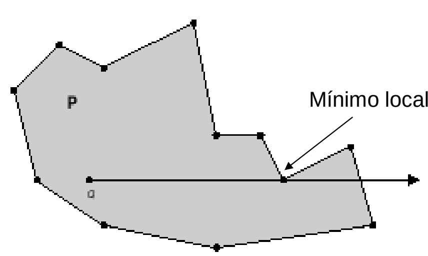


Par → OUT	
Ímpar → IN	

(a)

Problema Nº 1: Ponto em Polígono

(b) Se a **semi-recta passa por um vértice**, a intersecção **NÃO** deve ser considerada se o vértice é **máximo** (ou **mínimo**) **local**



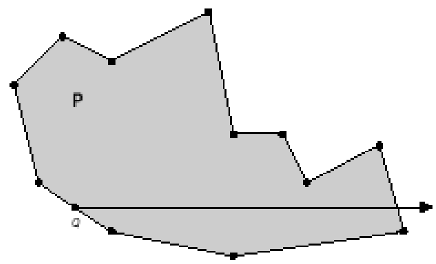
Par → OUT	
Impar → IN	

(b)

Problema Nº 1: Ponto em Polígono

(c) Se o **ponto *Q* pertence à fr(P)**, então ...

Par → OUT
Impar → IN

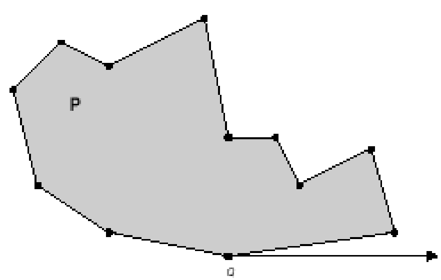


(c)

Problema Nº 1: Ponto em Polígono

(d) Se o **ponto *Q* coincide com um vértice de P**, então …

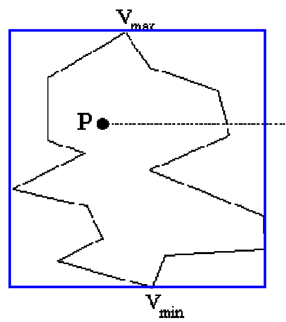
Par → OUT
Impar → IN



(d)

Problema Nº 1: Ponto em Polígono

- Acelerando o algoritmo…
  - **Teste de rejeição rápida (PinR):**
    - Comparar a posição do ponto *Q* em relação ao **rectângulo envolvente mínimo do polígono** (sem buracos), considerando que este tenha sido previamente determinado e armazenado.



Problema Nº 1: Ponto em Polígono

- O teste **PinR** tem complexidade ***O*(1)** e pode representar um ganho no desempenho importante.

```
Função PinR (Ponto Q, Ponto A, Ponto B): booleano
    Início
        Ponto C, D;
        C.x = min (A.x, B.x); C.y = min (A.y, B.y);
        D.x = max (A.x, B.x); D.y = max (A.y, B.y);
        Retorne ((Q.x >= C.x) & (Q.x <= D.x) &
                (Q.y >= C.y) & (Q.y <= D.y));
    Fim
```

Nota: A e B são vértices (não adjacentes) do rectângulo R

Problema Nº 1: Ponto em Polígono

- **PinR** em **PinP**

```
Função PinP (Ponto Q, Poligono P): booleano
  Início
    inteiro i, num_inter=0;
    Ponto C, D;
    Se (not(PinR(Q, A, B)) Então Retorne falso;
    Para i=0 até i<P.numVertices Faça
      Inicio
              C = P.v[i];
              D = P.v[i+1];
              Se (C.y ≠ D.y) Então
                Início
                   Calcular intersecção;
                   Se (intersecção for à direita de Q) Então
                        num_inter = num_inter+1;
                Fim
      Fim
      Retorne ((num_inter mod 2 <> 0);
  Fim
```


Problema Nº 1: Ponto em Polígono

A <u>convexidade</u> facilita ou não a localização dum ponto em relação a um polígono?

Sim!

Problema Nº 1: Ponto em Polígono (convexo)

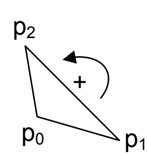
- **Preliminares:**
  - **Três pontos são colineares** se eles pertencem a uma mesma recta.
  - Em geometria cartesiana, prova-se que três pontos $p_o=(x_0, y_0)$, $p_1=(x_1, y_1)$ e $p_2=(x_2, y_2)$ são **colineares** sse:

$$\begin{vmatrix} 1 & x_0 & y_0 \\ 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \end{vmatrix} = 0$$

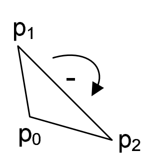
Problema Nº 1: Ponto em Polígono (convexo)

- **Preliminares:**
  - Verifica-se que se esses três pontos não são colineares, o **sinal desse mesmo determinante** é precisamente a orientação do triângulo, determinado por esses três pontos

$$\Delta(p_0, p_1, p_2) = \operatorname{sgn}\begin{vmatrix} 1 & x_0 & y_0 \\ 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \end{vmatrix} > 0, \ \textit{orientação positiva}$$

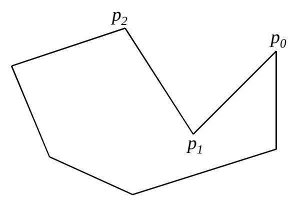


$$\Delta(p_0, p_1, p_2) = \operatorname{sgn}\begin{vmatrix} 1 & x_0 & y_0 \\ 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \end{vmatrix} < 0, \ \textit{orientação negativa}$$



Problema Nº 1: Ponto em Polígono (convexo)

- **Convexidade facilita a localização de pontos**
  - Vejamos agora como podemos determinar em **tempo sublinear (< $O(n)$)** se um ponto dado está no interior ou exterior de um **polígono convexo**
    - Um polígono simples P = (p_0, p_1,..., p_n) é chamado **polígono convexo** sse $\Delta(p_i, p_{i+1}, p_{i+2}) = +1$, para todo i, $0 \leq i \leq n\text{-}2$
    - Chamamos o **lado positivo** de um polígono convexo de interior do polígono e o **lado negativo** de exterior.



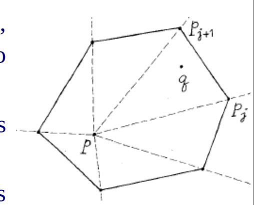
Não é convexo pois $\Delta(p_0, p_1, p_2) = -1$

Problema Nº 1: Ponto em Polígono (convexo)

Esboço do algoritmo que decide se um ponto ***q*** está no interior ou exterior de **P**

- Seja ***p*** um ponto no interior de **P**. Para cada vértice p_i de **P**, considere o segmento passando por p_i ligando ***p*** a um ponto no infinito na direcção do vector ($p_i - p$).
- Desta forma o plano fica **dividido em *n* "triângulos"** todos tendo ***p*** como vértice comum.
- Chamemos de T_i ao "triângulo" limitado pelos segmentos determinados por pp_i e pp_{i+1}. Como **estes "triângulos" estão ordenados circularmente em torno de *p*** pela própria sequência dos vértices de **P** dada, podemos **determinar por pesquisa binária** qual o "triângulo" dentro do qual o ponto de consulta ***q*** está.
- Finalmente:

  Se *q* está dentro de um "triângulo" T_i então *q* estará no interior de P sse *q* estiver do lado positivo da recta determinada pelo segmento p_ip_{i+1}



Problema Nº 1: Ponto em Polígono (convexo)

Algoritmo PinPConv(q, P)

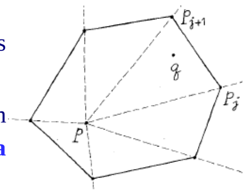
- Dado um **polígono convexo** **P** e um ponto q, devolve o lado de **P** em que q se encontra.

  1. Seja p o **baricentro** de algum triângulo formado por vértices de **P**.

  2. Como a ordem dos vértices $(p_0, p_1, \ldots, p_n)$ de **P** estabelecem uma **ordem cíclica** para as rectas pp_i, determinar por **pesquisa binária** o índice j $(0 \leq j \leq n\text{-}1)$ tal que:

$$\Delta(p,\ p_{j+1}, q) < 0 \;\&\&\; \Delta(p,\ p_j,\ q) \geq 0$$

  3. Devolver o sinal de $\Delta(p_j,\ p_{j+1}, q)$ **(+ in; - out)**

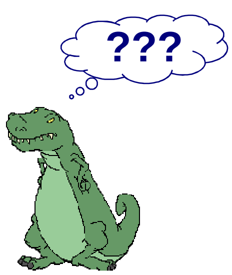


Complexidade:

Os passos **1** e **3** tomam tempo constante. A busca feita no passo **2** toma tempo logarítmico. Portanto, este algoritmo toma tempo total ***O(log n)***

Problema Nº1: Ponto em Polígono (estrelado)

Será possível generalizar o algoritmo PinPConv para polígonos estrelados quando se conhece algum ponto do seu núcleo?



Problema Nº 2: Área de Polígonos

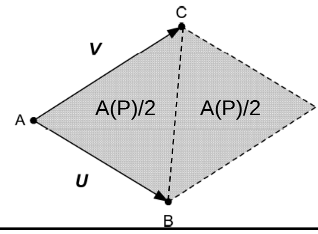
- **Objectivo:** estudo de algoritmos para o cálculo de áreas de polígonos simples.

(1) Área (A) de um triângulo T(A, B, C)

- A área de um triângulo é igual a metade do produto entre sua base e sua altura.
- Sabe-se que a norma do vector **UxV**, denotada por:

  |UxV| = |U||V|sen(θ) e que

  |UxV| é igual a área do paralelogramo com lados U e V, onde **θ** é o ângulo do vector U ao V.
- **|UxV|** é igual a duas vezes a área do triangulo definido por eles.



Problema Nº 2: Área de Polígonos

- O **produto vectorial** pode ser calculado a partir do seguinte determinante:

$$\begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ x_U & y_U & z_U \\ x_V & y_V & z_V \end{vmatrix} = (y_U z_V - z_U y_V)\hat{i} + (z_U x_V - x_U z_V)\hat{j} + (x_U y_V - y_U x_V)\hat{k}$$

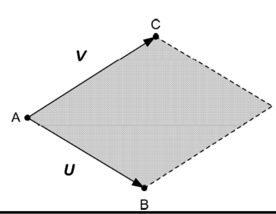
onde *i*, *j* e *k* são os vectores unitários nas direcções x, y e z respectivamente.

- Como se está tratando de vectores bidimensionais temos $Z_U = Z_V = 0$ e, portanto a área A do triângulo T:

$$A(T) = 1/2(X_U Y_V - Y_U X_V)$$

- Mas na realidade, U = B - A, e V = C - A Portanto, a expressão acima pode ser reescrita como

$$A(T)=1/2(X_A Y_B - Y_A X_B + Y_A X_C - X_A Y_C + X_B Y_C - Y_B X_C)$$



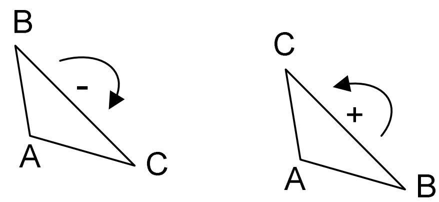
Problema Nº 2: Área de Polígonos

- A área do triângulo pode ser calculada pela expressão

$$A(T)=1/2(X_AY_B - Y_AX_B + Y_AX_C - X_AY_C + X_BY_C - Y_BX_C)$$

será:

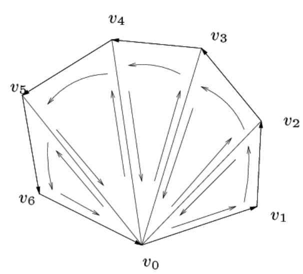
- **positiva** se os vértices A, B e C formarem um circuito em **sentido anti-horário (CCW)**
- **negativa** se formarem um circuito no **sentido horário (CW)**
- a área será **exactamente zero** se os três vértices estiverem **alinhados**



Problema Nº 2: Área de Polígonos

(2) Área de um Polígono Convexo

- Para calcularmos a área de um **polígono convexo** P com vértices v_0, v_1, ..., v_n, podemos primeiro **triangularizar** P e depois somar as áreas de cada um dos triângulos.



- Então a área sinalizada A(P) de um polígono convexo é dada por:

$$A(P) = A(v_0, v_1, v_2) + A(v_0, v_2, v_3) + ...+ A(v_0, v_{n-1}, v_n)$$

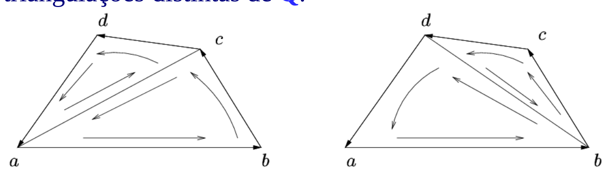
onde os índices são tomados por módulo *n*

- **É de esperar que a área sinalizada de P não depende da triangulação**

Problema Nº 2: Área de Polígonos

□ Área de um Polígono Convexo

- Consideremos um quadrilátero convexo **Q = (*a*, *b*, *c*, *d*).** Existem duas triangulações distintas de **Q**.



- Agora temos que

$$\mathcal{A}(Q) = \mathcal{A}(a,b,c) + \mathcal{A}(a,c,d) = \mathcal{A}(d,a,b) + \mathcal{A}(d,b,c).$$

Se $a = (a_x, a_y), b = (b_x, b_y), c = (c_x, cy), d = (d_x, d_y)$ então

$$\begin{aligned} 2\mathcal{A}(Q) &= a_x b_y - a_y b_x + a_y c_x - a_x c_y + b_x c_y - b_y c_x \\ &\quad + a_x c_y - a_y c_x + a_y d_x - a_x d_y + c_x d_y - d_x c_y. \end{aligned}$$

- Vemos que vários termos aparecem em ***A*(*a*, *b*, *c*)** e em ***A*(*a*, *c*, *d*)** com sinais trocados pelo que podemos perceber que termos correspondentes à diagonal ***ac*** se cancelam

Problema Nº 2: Área de Polígonos

□ Área de um Polígono Convexo

- **Assim generalizando teremos dois termos por aresta do polígono e nenhum termo por diagonal.**
- Portanto se $P = (v_0, v_1, ..., v_{n-1})$ é um **polígono convexo**, onde $v_i = (x_i, y_i)$ ($i = 0,..., n-1$), então a área sinalizada ***A*(*P*)** de *P* satisfaz,

$$A(P) = \frac{1}{2}\sum_{i=0}^{n-1}(x_i y_{i+1} - y_i x_{i+1})$$

onde os índices são tomados por módulo *n*

Problema Nº 2: Área de Polígonos

Este resultado pode ser generalizada para polígonos simples arbitrários?

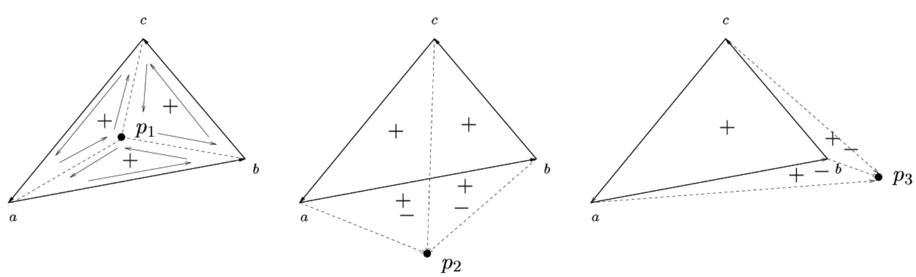
Sim!

Problema Nº 2: Área de Polígonos

(3) Teorema da Área de Polígonos

- Seja **T(*a*, *b*, *c*)** um triângulo com vértices orientados positivamente. Seja ***p*** um **ponto arbitrário** do plano. Afirmamos que:

  A(T) = A(*p*, *a*, *b*) + A(*p*, *b*, *c*) + A(*p*, *c*, *a*)



Lema 1: *Seja T(a, b, c) um triângulo orientado positivamente e p um ponto qualquer do plano. Então*

A(T) = A(*p*, *a*, *b*) + A(*p*, *b*, *c*) + A(*p*, *c*, *a*)

Problema Nº 2: Área de Polígonos

- **Teorema da Área de Polígonos**

Teorema: *Seja P = (v_0, v_1, ..., v_{n-1}) um polígono simples com vértices orientados positivamente e seja p um ponto qualquer do plano. Então*

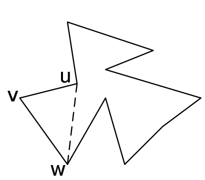
(1) $$A(P) = A(p, v_0, v_1) + A(p, v_1, v_2) + \ldots + A(p, v_{n-1}, v_0)$$

ademais, se $v_i = (x_i, y_i)$ $(i = 0, \ldots, n\text{-}1)$, *então*

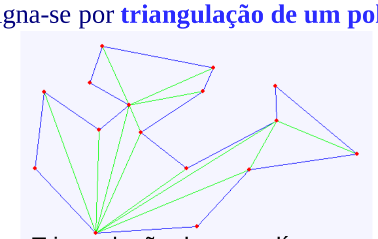
(2) $$A(P) = \frac{1}{2}\sum_{i=0}^{n-1}(x_i y_{i+1} - y_i x_{i+1})$$

Problema Nº 2: Área de Polígonos

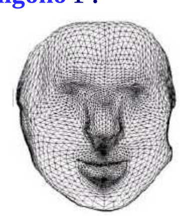
- **Teorema da Área de Polígonos**
  - Para provar este Teorema vamos precisar do seguinte:
    - Uma **diagonal** de um polígono P é um segmento de recta entre dois vértices de P que se vêem claramente.
    - Diremos que 3 vértices consecutivos ***u***, ***v***, e ***w*** de um polígono formam uma **orelha** se o segmento de recta ***uw*** é uma diagonal
    - A *decomposição* de um polígono P em *triângulos* por colocação do *maior número* possível de **diagonais** que, duas a duas, não se intersectam designa-se por **triangulação de um polígono** P.



uw é um diagonal



Triangulação de um polígono



Problema Nº 2: Área de Polígonos

- Teorema da Área de Polígonos
  - **Teorema:** (**Meister's Two Ears Theorem**) - *Todo polígono com pelo menos 4 vértices possui pelo menos duas orelhas que não se sobrepõem.*
  - **Teorema** (**Triangulação vs orelhas**) - *Seja P um polígono com pelo menos 4 vértices e T uma triangulação de P. Então pelo menos dois triângulos de T formam orelhas de P*

    Prova: (será estuda posteriormente…)

Problema Nº 2: Área de Polígonos

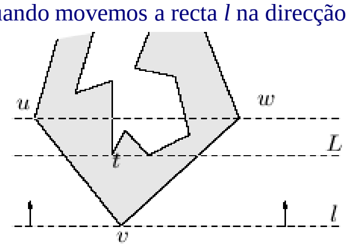
- **Teorema da Área de Polígonos**

Lema: (**Meister, 1975**) - *Todo polígono com pelo menos 4 vértices possui uma diagonal.*

Prova:

Seja P um polígono com $n \geq 4$ vértices e seja v um vértice convexo de P. Sejam u e w vértices adjacentes de v. Então se uw é uma **diagonal** → não há o que provar. Logo, suponhamos que uw não é uma diagonal de P, ou seja $uw \not\subset P$.

Como $n \geq 4$ → $\Delta(v, u, w)$ contém pelos menos um vértice de P distinto de v, u e w. Seja t um vértice ($t \subset \Delta(v, u, w)$) "mais próximo" de v. Logo, t é o primeiro vértice de P atingindo quando movemos a recta l na direcção uw.



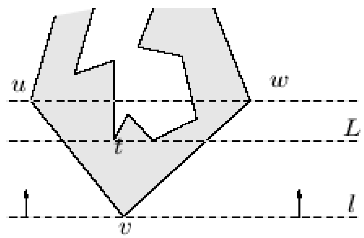
Problema Nº 2: Área de Polígonos

- **Teorema da Área de Polígonos**

Lema: (Meister, 1975) - *Todo polígono com pelo menos 4 vértices possui uma diagonal.*

Prova: (**continuação**)

Afirmamos que vt é uma diagonal de P. De facto, seja L a recta passando por t e paralela a uw. Notemos que a intersecção do semi-plano determinado por L contento o vértice v com o $\Delta(v, u, w)$ é um triângulo que não contêm nenhum ponto de fr(P) no seu interior. Logo, o vértice v vê t claramente. Portanto, vt é uma diagonal de P. ♦



Problema Nº 2: Área de Polígonos

□ **Teorema da Área de Polígonos**

Prova (do teorema da área de polígonos):

Por indução no número de vértices de P.

Se P tem três vértices então o resultado segue do **Lema 1**

Suponhamos que **(1)** é verdadeira para polígonos com até n-1 vértices, $n \geq 4$, e suponhamos que P é um polígono com n vértices.

Pelo **Teorema das duas Orelhas de Meister** temos que P tem uma orelha. Vamos renumerar os vértices de P, se necessário de tal forma que os vértices v_{n-2}, v_{n-1},v_0 formem uma orelha (i.e. um triângulo) designado por E. Seja P' o polígono resultante de P após a remoção de E. Pela hipótese de indução temos que:

$\mathbf{A(P') = A(p, v_0, v_1) + A(p, v_1, v_2) + ... + A(p, v_{n-1}, v_0)}$

Ademais pelo **Lema 1** temos que

$\mathbf{A(E) = A(p, v_{n-2}, v_{n-1}) + A(p, v_{n-1}, v_0) + A(p, v_0, v_{n-2})}$

Então temos que: **A(P) = A(P') + A(E)**

$\mathbf{A(P) = A(p, v_0, v_1) + \ldots + A(p, v_{n-3}, v_{n-2}) + A(p, v_{n-2}, v_0) +}$

$\mathbf{A(p, v_{n-2}, v_{n-1}) + A(p, v_{n-1}, v_0) + A(p, v_0, v_{n-2})}$

Portanto, como $\mathbf{A(p, v_0, v_{n-2}) = - A(p, v_{n-2}, v_0)}$ temos que a equação **(1)** se verifica.

A equação **(2)** pode ser obtida através da simples expansão dos termos em **(1)** ♦

Problema Nº 3: Centróide de um Polígono

- O centro de gravidade ou centro de massa, mais conhecido como **centróide** de um polígono pode ser obtido da seguinte forma:
  - **Particionar o polígono em triângulos**
  - **Calcular a média ponderada dos centros de gravidade dos triângulos usando suas áreas como peso**
- O centro de gravidade de cada triângulo é simplesmente a média das coordenadas de seus vértices, ou seja, para um T(A, B, C):

$$x_G = \frac{x_A + x_B + x_C}{3} \quad e \quad y_G = \frac{y_A + y_B + y_C}{3}$$

- **Dificuldade:** é preciso implementar um algoritmo de triangulação de polígonos.

Problema Nº 3: Centróide de um Polígono

- Nessa estratégia os **centróides** dos triângulos são combinados usando um processo de média ponderada pela área.
- O **centróide de um polígono** formado por dois triângulos T_1 e T_2, cujos centróides são, respectivamente, (x_{G1}, y_{G1}) e (x_{G2}, y_{G2}) é o ponto (x_G, y_G), onde

$$x_G = \frac{x_{G1}A(T_1) + x_{G2}A(T_2)}{A(T_1) + A(T_2)} \quad e \quad y_G = \frac{y_{G1}A(T_1) + y_{G2}A(T_2)}{A(T_1) + A(T_2)}$$

- Então o **centróide do polígono pode ser determinado de maneira incremental, adicionando um triângulo e seu centróide por vez e calculando as coordenadas do centróide do conjunto.**

Problema Nº 3: Centróide de um Polígono

- **Uma estratégia mais simples:**
  - O mesmo processo de média ponderada pela área pode ser usado, considerando todos os triângulos formados entre um ponto fixo, por exemplo (0, 0), e cada par de vértices sucessivos, (v_i, v_{i+1}).
  - Então aplicando os resultados do teorema da área de polígonos e a fórmula para determinar o **centróide** de um polígono obtemos que:

$$x_C = \frac{\sum_{i=0}^{n-1}(x_{i+1}+x_i)\times(x_i y_{i+1} - y_i x_{i+1})}{3A(P)} \qquad y_C = \frac{\sum_{i=0}^{n-1}(y_{i+1}+y_i)\times(x_i y_{i+1} - y_i x_{i+1})}{3A(P)}$$

$$A(P) = \frac{1}{2}\sum_{i=0}^{n-1}(x_i y_{i+1} - y_i x_{i+1})$$

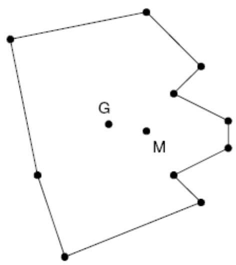
  - **Atenção: Utilizando este método para calcular o centróide de um polígono não é preciso determinar a triangulação do polígono e a sua complexidade é *O*(*n*)**

Problema Nº 3: Centróide de um Polígono

- **Algumas observações:**
  - Apesar da simplicidade do processo, **não existe garantia de que o centróide será um ponto pertencente ao polígono**.
  - Caso seja necessário encontrar um ponto interno a um polígono simples dado, pode-se utilizar um processo que consiste precisamente em identificar rapidamente uma diagonal do polígono (O'Rourke, 1998).
  - O **centróide** pode ser determinado também por outros métodos:
    - como o centro do rectângulo envolvente mínimo ou,
    - como o centro de um círculo inscrito ou circunscrito ao polígono.

Problema Nº 3: Centróide de um Polígono

- **Algumas observações:**
  - Nos **GIS** (*Geographics Information Systems*) para determinar o centróide utilizam frequentemente o seguinte método: obter a média aritmética (**M**) das coordenadas x e y dos vértices do polígono.
  - Este método é muito simples, mas a existência de alguma concentração de vértices numa região do polígono causa um deslocamento indesejável do **centróide**.



O **centróide G**, é calculado pela média ponderada das áreas dos triangulo,
O **centróide M** é calculado pela média aritmática das coordenas dos vértices do polígono.

Problema Nº 3: Centróide de um Polígono

- **Algoritmo de O'Rourke, 1998:**
  - Identificar um vértice convexo v_i (por exemplo, o vértice inferior mais à direita).
  - Para cada outro vértice v_j do polígono verificar:
    - se v_j estiver dentro do $T(v_{i-1}, v_i, v_{i+1})$, então
      - calcular a distância (euclidiana) $d(v_i,v_j)$
      - armazenar v_j em q se esta distância for um novo mínimo
  - Se algum ponto interior a $T(v_{i-1}, v_i, v_{i+1})$, for encontrado, então
    - o ponto médio do segmento qv_i é interior ao polígono
  - senão,
    - o ponto médio do segmento $v_{i-1}v_{i+1}$ é interior ao polígono.

Material Complementar

- O **Teorema de Pick** e o cálculo da área de polígonos:
  http://cmup.fc.up.pt/cmup/pick/pick2.html

- **Outros recursos**
  - **Polygon Area and Centroid**
  http://www.efg2.com/Lab/Graphics/PolygonArea.htm