



# Protótipo de um editor gráfico 2D para palm

**Aluno: Eduardo Hildebrandt**

**Orientador: Dr. Paulo César  
Rodacki Gomes**

# Tópicos

- **Introdução**
- **Geometria computacional**
- **Algoritmo de ponto em Polígono**
- **Intersecção de Semi-retas**
- **Tolerância numérica**
- **Desenvolvimento do Protótipo**
- **Ferramentas utilizadas**
- **Conclusão**

# Introdução

- **Em 1996 a Palm Inc. lança o primeiro computador de mão.**
- **Novos modelos coloridos são lançados**
- **Exploração dos recursos gráficos da plataforma**
- **Estudo de ferramenta de desenvolvimento**

# Objetivos

- **Editor gráfico, que permita desenhar e selecionar polígonos**
- **Implementação do Algoritmo de ponto em polígono**

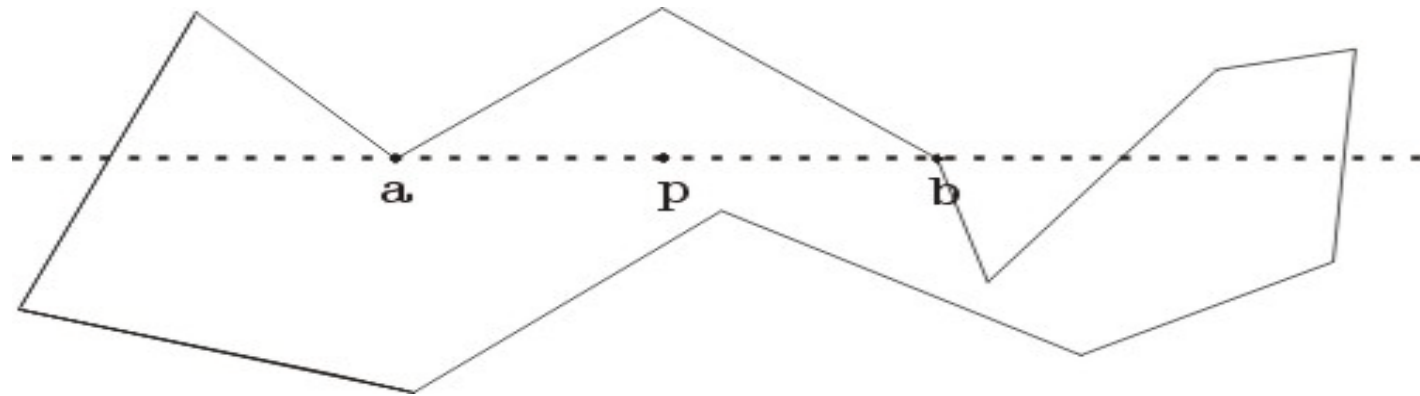


# Geometria Computacional

- **Estudar problemas geométricos sob o ponto de vista algorítmico, focando o interesse em solucionar um problema utilizando o menor número possível de operações elementares de modo a trazer eficiência no cálculo da solução. (Freitas, 2003)**




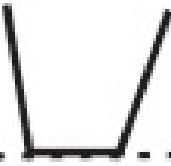


# Algoritmo de ponto em Polígono

- **Identificação da inclusão ou exclusão de um ponto em um polígono**



# Algoritmo de ponto em Polígono

- **Casos especiais que devem ser tratados**

Casos						
Número de Interseções	0	2	1	0	2	1

# Algoritmo de ponto em Polígono

**dados:**  $p = (x_0, y_0)$ ;  $p_i = (x_i, y_i)$ ,  $i = 1, \dots, n$ ;  $p_{n+1} = p_1$ .

**inicialização:**  $N = 0$ ;

**para**  $i = 1..n$

**se**  $y_i \neq y_{i+1}$  **entao**

$(x, y) =$  a intersecção de  $p_i p_{i+1}$  com  $L$

**se**  $x = x_0$  **entao**

$p_0$  é da fronteira: PARE.

**senao**, **se**  $x > x_0$  e  $y > \min(y_i, y_{i+1})$  **entao**

$N = N + 1$ .

**senão**, **se**  $p$  pertence ao lado horizontal  $p_i p_{i+1}$  **entao**

$p$  é da fronteira: PARE.

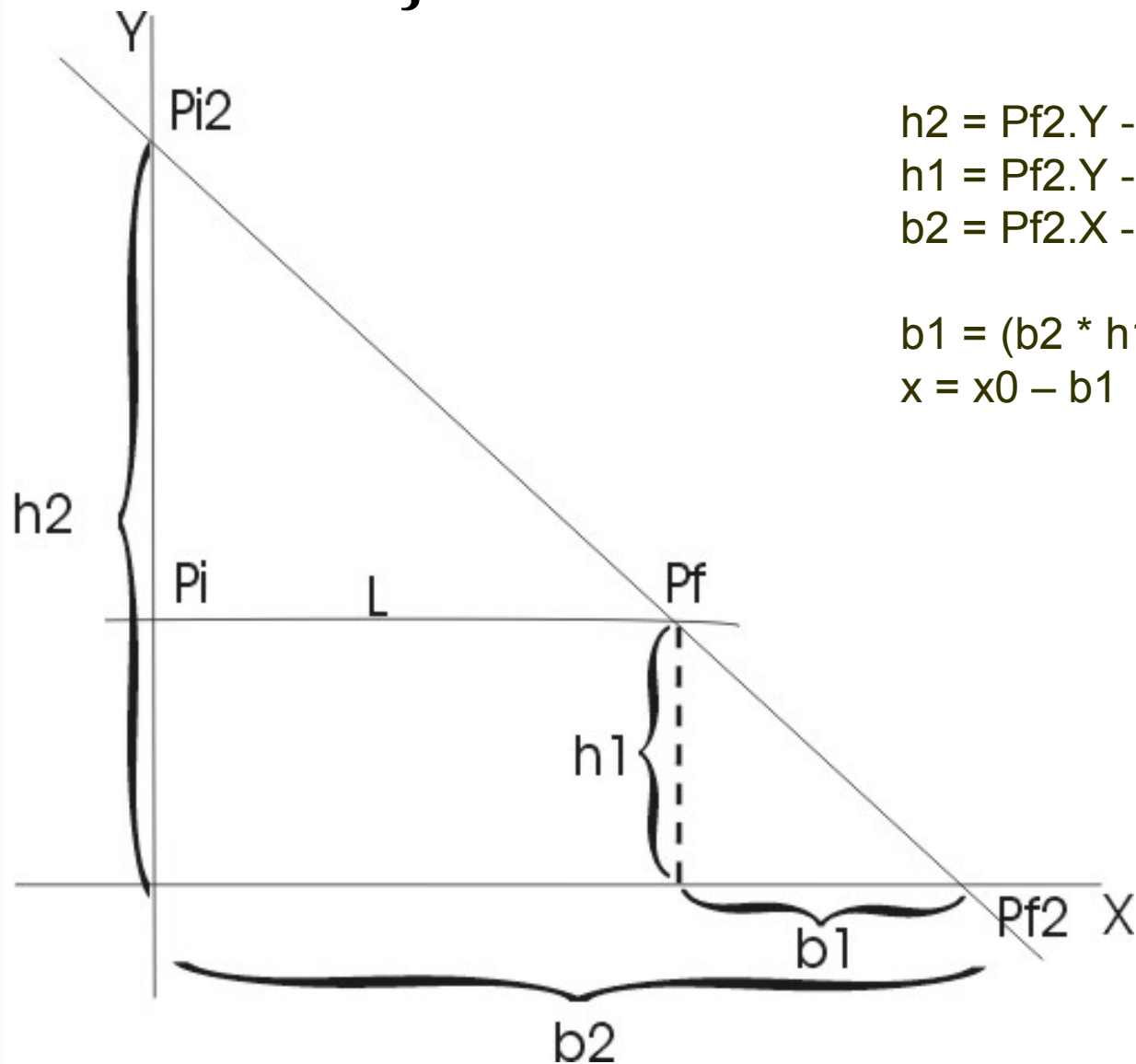
**se**  $N$  é impar, **então**  $p$  é interior; **senão**,  $p$  é exterior.



# Intersecção de Semi-retas

- **Auxilio ao algoritmo de ponto em polígono**
- **Um das semi-retas horizontal**
- **Semelhança de triângulos**
- **Segundo Giovanni (1988), dois triângulos são semelhantes se, e somente se, têm dois ângulos respectivamente congruentes.**

# Intersecção de Semi-retas



$$h_2 = P_{f2}.Y - P_{i2}.Y;$$

$$h_1 = P_{f2}.Y - P_i.Y;$$

$$b_2 = P_{f2}.X - P_{i2}.X;$$

$$b_1 = (b_2 * h_1) / h_2$$

$$x = x_0 - b_1$$

# Tolerância numérica

- **Dificuldade de representar a precisão numérica necessária**
- **Limitação da plataforma computacional**
- **Facilitar a interação do usuário com o programa**

# Estrutura de dados

- **Segundo Foley (1990), as estruturas de dados para a geometria computacional são um grupo lógico de primitivas, atributos e outras informações, que podem representar a geometria, aparência ou algum dado relevante ao software**
- **Separação das informações geométricas e topológicas**

# Plataforma Palm

- **Computadores de mão, “palmtop”**
- **Visam a apresentação de informações em qualquer lugar e qualquer momento**
- **Interação através de uma “caneta”**
- **Botões de acesso rápido**

# Palm m130



**Palm m130  
handheld in  
USB  
HotSync®  
Cradle**

# Palm OS

- **Sistema operacional dos palmtop's**
- **É multi-tarefa, mas a interface limita a um programa por vez**
- **Interface simples**
- **Memória RAM e ROM são a mesma fisicamente**

# Programa Palm

- **Estrutura de programa semelhante a um programa para plataforma PC Win32**
- **Comunicação do sistema operacional com o programa através de eventos**





# Programa Palm

## **Estrutura de um programa:**

- **Inicialização do programa**
- **Loop de eventos**
- **Tratadores de eventos**

# Emulador

- Programa que emula um computador palm
- Necessita de uma “rom” que é a imagem do sistema operacional
- Permite a depuração do programa

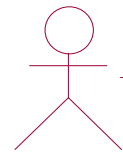
# Emulador



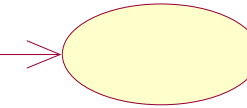
# Especificação

- **Orientação a Objetos, utilizando *UML* com a ferramenta *Rational Rose***

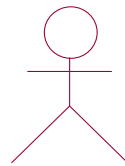
# Especificação



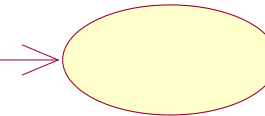
Usuario



Criar Poligono



Usuario

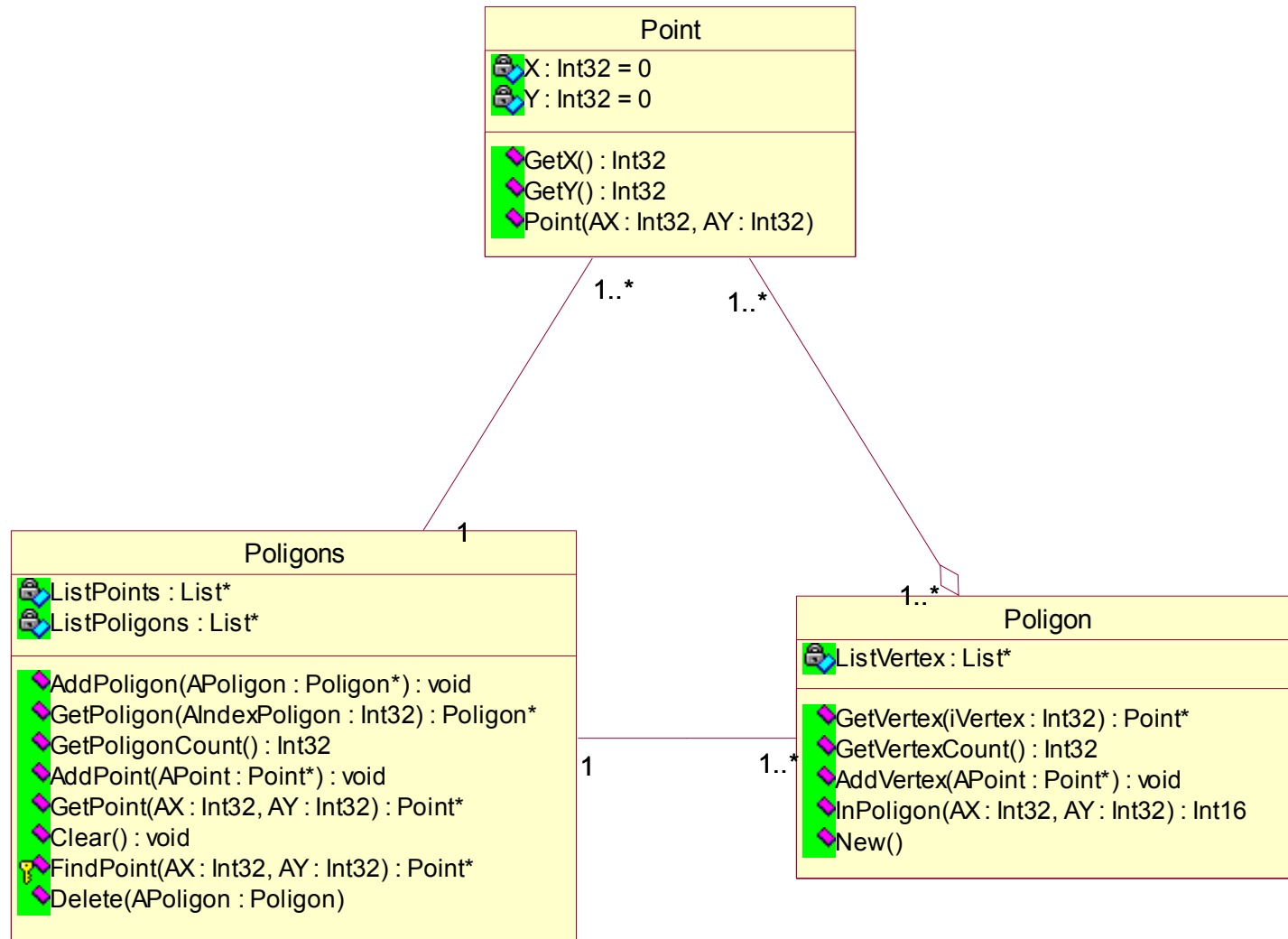


Selecionar Poligono

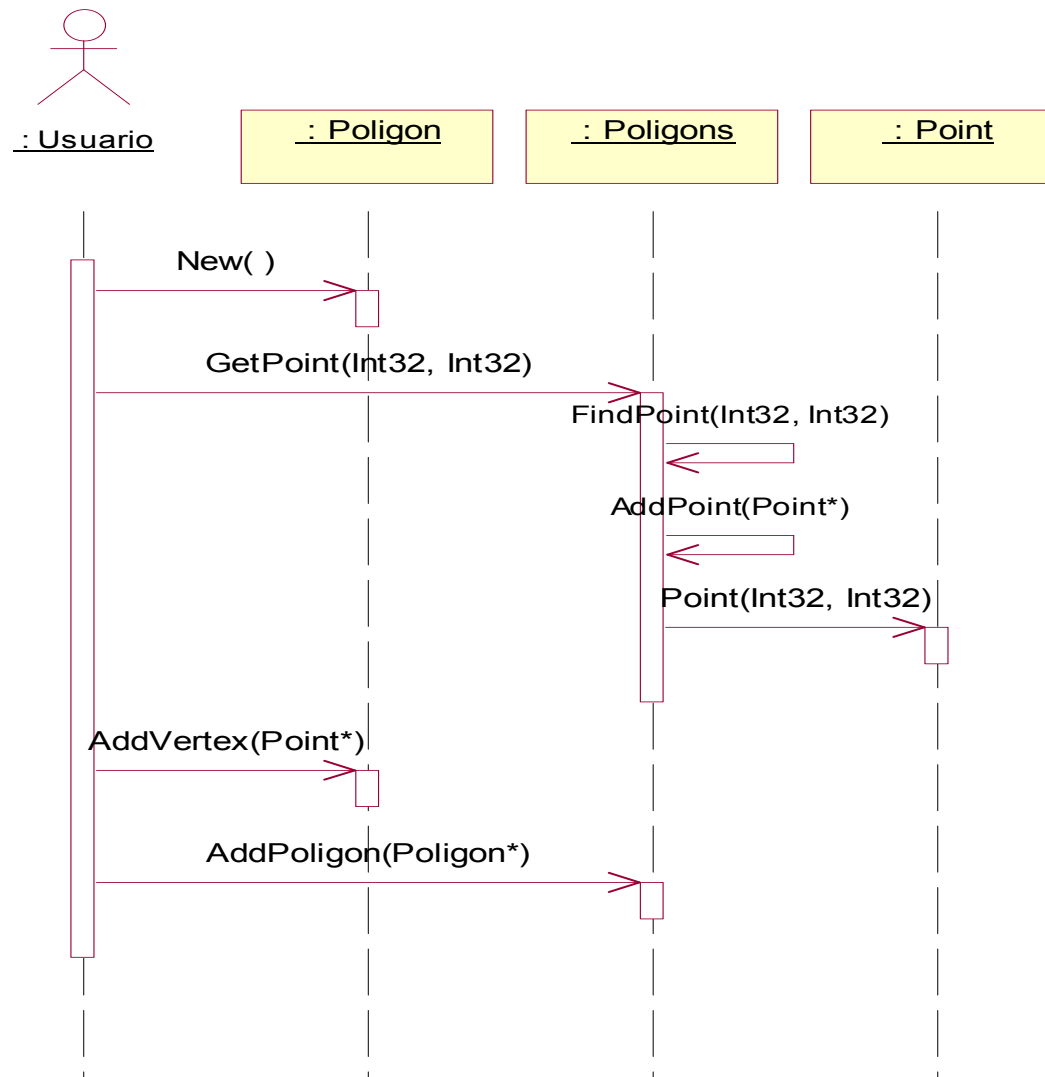
# Especificação



# Especificação

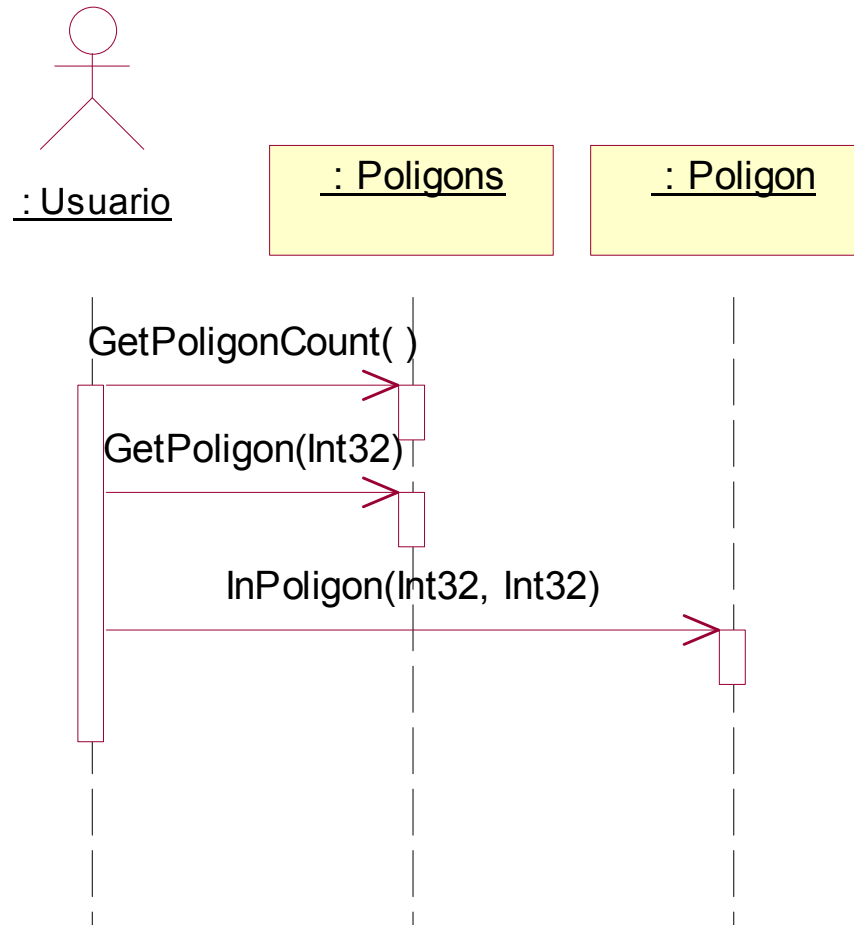


# Especificação “criar polígono”





# Especificação “Selecionar”



# Protótipo

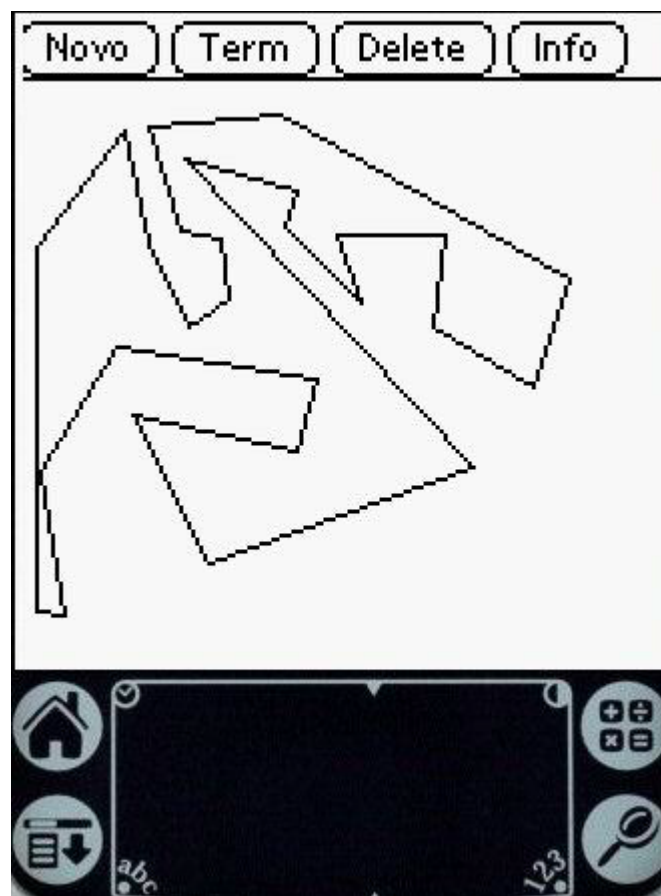
- **Programação Orientada a Objetos**
- **Compilador *gcc* para palm**
- **Editor de texto *vi***
- **Depurador *gdb***

# Protótipo

**O gcc para palm é dividido em:**

- **Compilador gcc compila o código do programa**
- **Compilador pilrc compila as interfaces do programa**
- **Compilador build-prc gera o executável final do palm**

# Operacionalidade



# Conclusões

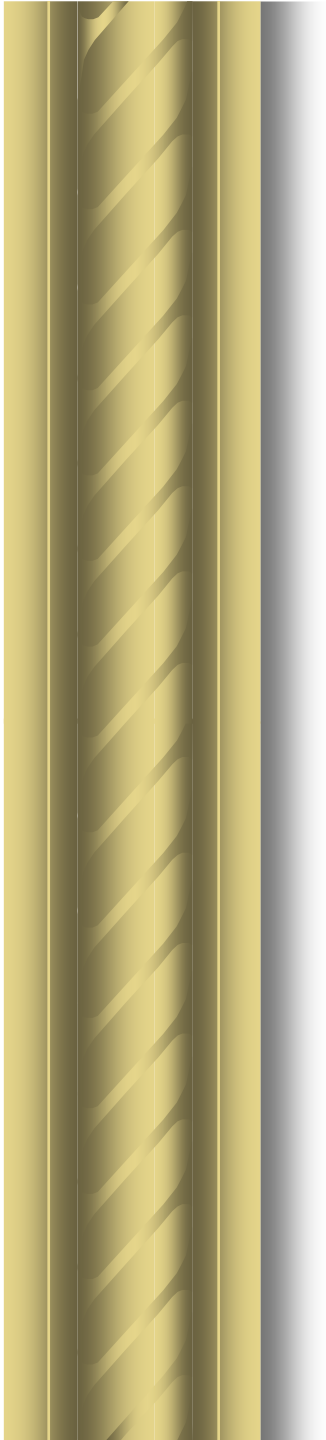
- **Investigação dos recursos gráficos 2D no Palm**
- **Investigação do processo de desenvolvimento no Palm**

# Conclusões

- **Desenvolvimento em plataforma linux utilizando o gcc e um emulador**
- **Protótipo**
  - **Algoritmo de ponto em polígono é simples e eficiente...**
  - **... porém, com o aumento da quantidade de polígonos, há uma queda no desempenho**

# Extensões

- **Explorar os demais recursos do palm (som, imagens, etc)**
- **Melhorar o desempenho do algoritmo de seleção (NEDEL, 2003)**
- **Explorar recurso de sincronização**
- **Análise mais detalhada do compilador gcc**



**Obrigado**