

Introdução à Engenharia

ENG1000

Aula 11 – Utilizando Imagens na PlayLib

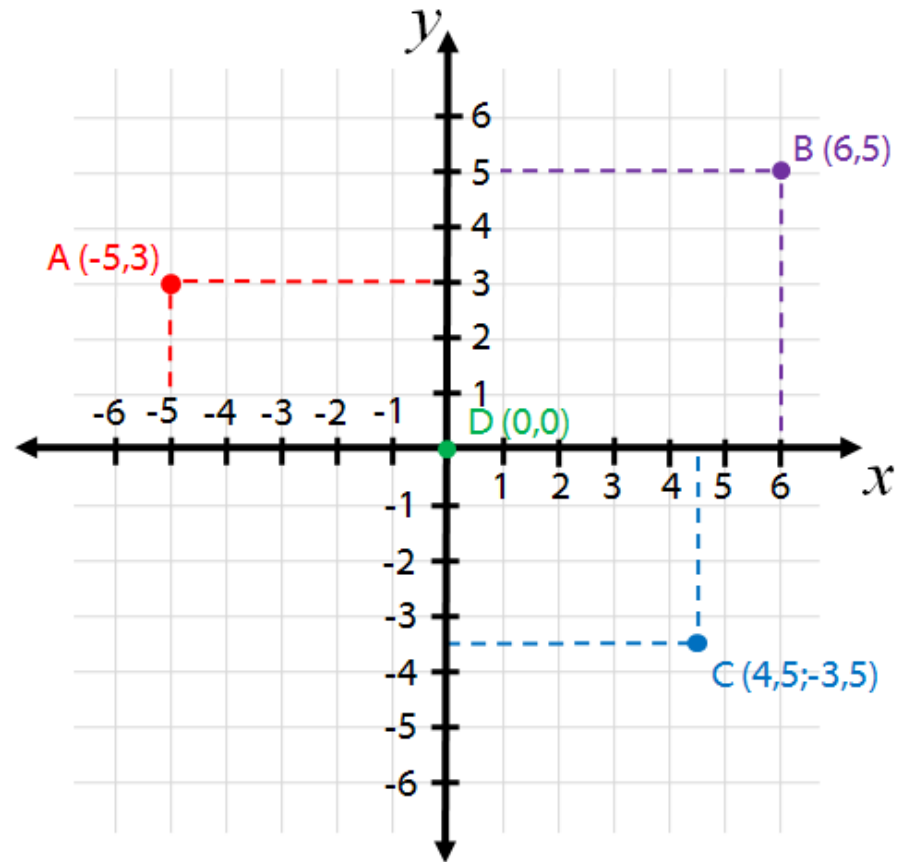
Prof. Augusto Baffa
< abaffa@inf.puc-rio.br >

Biblioteca Gráfica - PlayLib

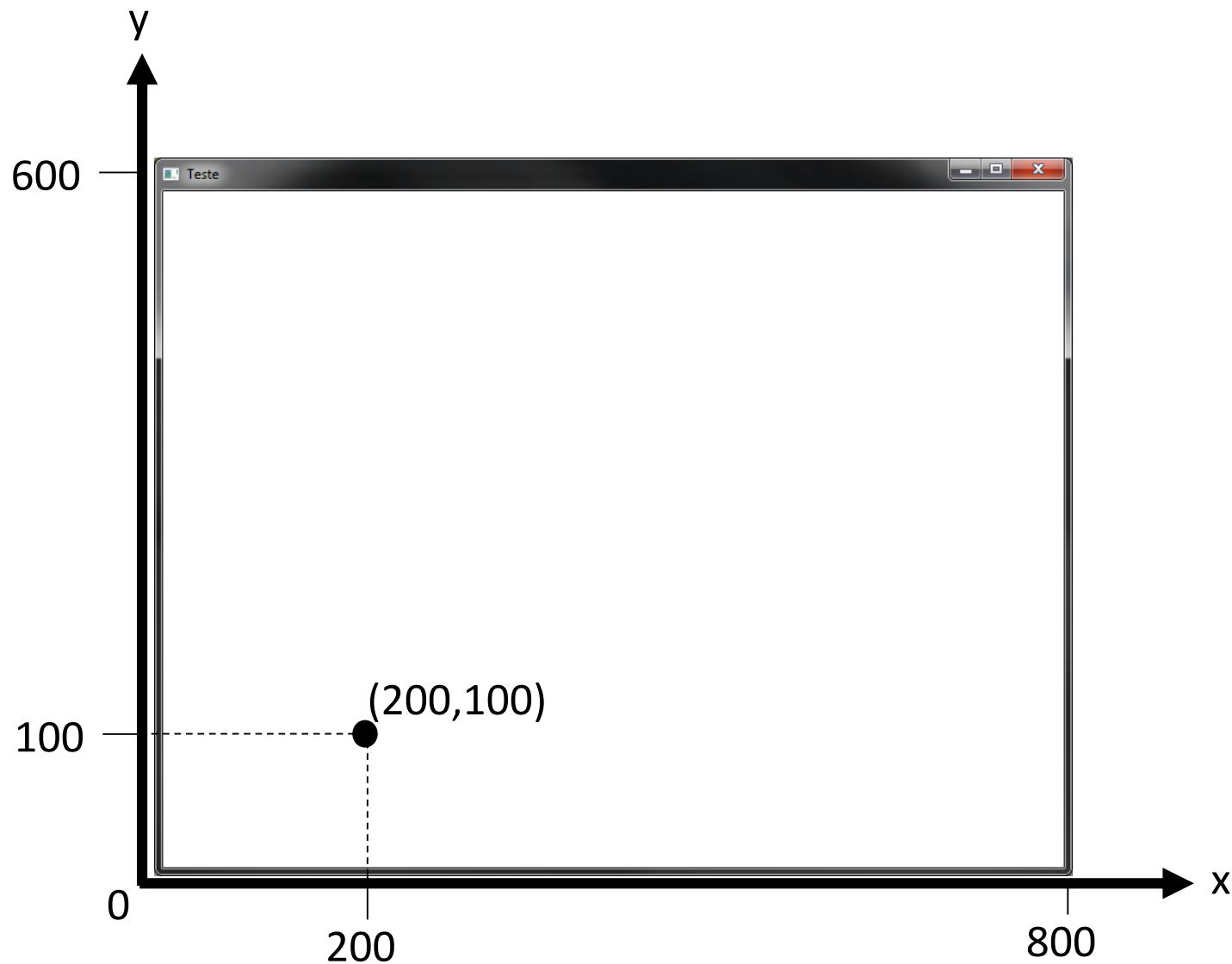
- **Conjunto de funções** para criação e manipulação de formas geométricas, imagens, áudio, janelas...
- Baseada na API **OpenGL**.
- Pode ser usada para criação de **jogos 2D, simulações, animações** e outros aplicativos.
- **Desenvolvida especialmente para esse curso!**

Coordenadas de Tela

- Sistema de Coordenadas Cartesiano
- Duas dimensões (2D)
- Coordenadas X e Y



Coordenadas de Tela



Desenhando Imagens

- **Para desenhar uma imagem na tela é necessário:**
 - **(1)** Criar uma variável do tipo **Image**.
 - **(2)** Carregar a imagem do arquivo usando o comando **LoadPNGImage**.
 - **(3)** Desenhar efetivamente a imagem na tela usando o comando **DrawImage2D**.

Desenhando Imagens

- **(1) Criar uma variável do tipo Image:**

```
Image minha_imagem;
```

OBS: Sempre declare as variáveis Image como **variáveis globais**.

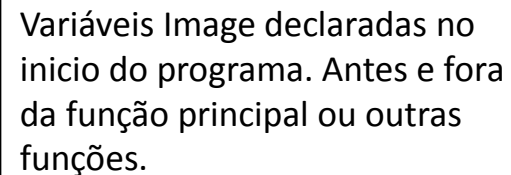
Exemplo:

```
#include "Graphics.h"  
using namespace GraphicsLib;
```

```
Graphics graphics;  
Image minha_imagem1;  
Image minha_imagem2;
```

```
int main(void)  
{  
...  
}
```

Variáveis Image declaradas no início do programa. Antes e fora da função principal ou outras funções.



Desenhando Imagens

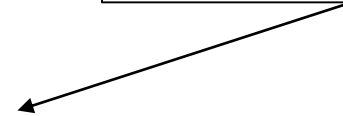
- (2) Carregar a imagem do arquivo usando o comando LoadPNGImage:

```
minha_imagem.LoadPNGImage("Mario.png");
```

Exemplo:

```
int main(void)
{
...
    minha_imagem.LoadPNGImage("Mario.png");
...
}
```

Carrega a imagem do arquivo **Mario.png** para a variável `minha_imagem`.



OBS: Cada imagem deve ser carregada **apenas uma vez**. Por isso, nunca carregue a imagem diretamente de dentro do Loop Principal.

Desenhando Imagens

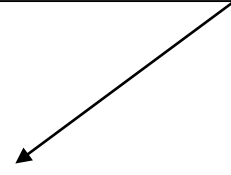
- **(3) Desenhar efetivamente a imagem na tela usando o comando DrawImage2D.**

```
graphics.DrawImage2D(200, 200, 256, 256, minha_imagem);
```

Exemplo:

```
void MainLoop()  
{  
...  
    graphics.DrawImage2D(200, 200, 256, 256, minha_imagem);  
...  
}
```

Desenha a imagem "minha_imagem" na posição (200, 200) com tamanho (256, 256) na tela.



Desenhando Imagens

- **Também é possível definir a posição e tamanho das imagens em variáveis armazenadas dentro do objeto Image. Para isso, deve-se:**

- 1) Criar uma variável do tipo Image:

```
Image minha_imagem;
```

- 2) Carregar a imagem do arquivo usando o comando LoadPNGImage:

```
minha_imagem.LoadPNGImage("Mario.png");
```

- 3) Definir a posição da imagem com o comando SetPosition:

```
minha_imagem.SetPosition(100,100,256,256);
```

- 4) Desenhar a imagem na tela com o comando DrawImage2D:

```
graphics.DrawImage2D(minha_imagem);
```

Desenhando Imagens

- Carregando uma Imagem:

```
void Image.LoadPNGImage(char *filename);
```

Exemplo:

```
Image mario;  
mario.LoadPNGImage("Mario.png");
```

Declaração da variável
do tipo Image que vai
armazenar a imagem

Carrega o arquivo
"Mario.png" para a
variável "mario"



Desenhando Imagens

- **Desenhando Imagens na Tela:**

```
void DrawImage2D(int x, int y, int width, int height, Image image);  
void DrawImage2D(Image image);  
void DrawImage2D(int x, int y, int width, int height, int crop_x,  
                 int crop_y, int crop_width, int crop_height, Image image);
```

Exemplo:

```
graphics.DrawImage2D(200, 200, 256, 256, mario);
```



Desenha a imagem "mario" na posição (200, 200) com tamanho (256, 256) na tela.

Desenhando Imagens

- **Definindo a Posição uma Imagem:**

```
void Image.SetPosition(int x, int y, int width, int height);
```

Exemplo:

```
mario.SetPosition(200, 200, 256, 256);
```



Define a posição da imagem
"mario" na posição (200, 200)
com tamanho (256, 256)

Desenhando Imagens

- **Observações importantes sobre imagens:**
 - **Somente são aceitas imagens no formato PNG.** Mas isso não é uma limitação, o formato PNG é um dos melhores formatos para esse tipo de aplicação. A principal vantagem é que ele permite o uso de **transparência** nas imagens.
 - Cerifique-se de que as imagens que serão lidas estão **dentro da pasta do seu projeto do Visual Studio**. Se preferir armazená-las em outras pastas você deve fornecer o caminho completo para o diretório onde as imagens estão para o comando LoadPNGImage.
 - Se a sua imagem estiver em **outro formato** (JPG, GIF, BMP...) você deve convertê-la para o formato PNG antes de carregá-la.

Exercícios

Lista de Exercícios 07 – Imagens e Interação

<http://www.inf.puc-rio.br/~abaffa/eng1000/>

Enviar para abaffa@inf.puc-rio.br até 13/abril

Assunto: ENG01000 Ex07

Não Esquecer de colocar nome e matrícula.