

Configuração da Janela

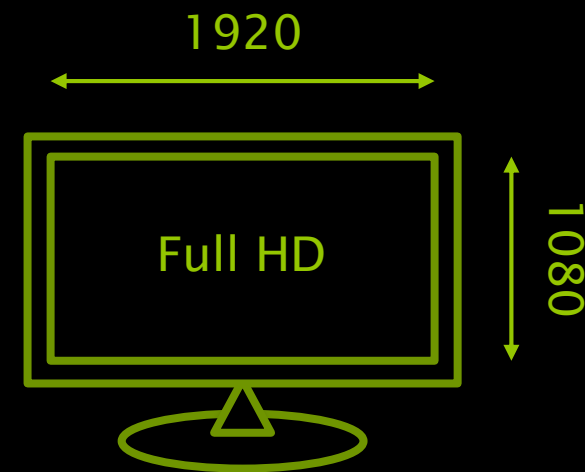
Programação de Jogos

Judson Santos Santiago

Introdução

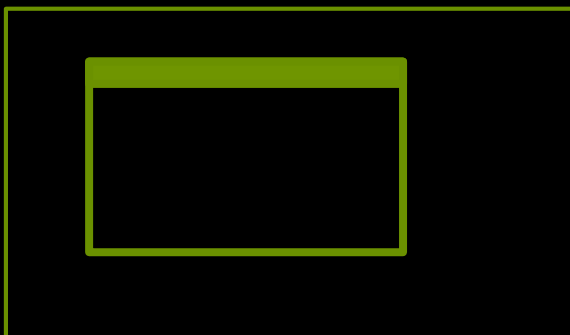
- ▶ Um jogo é **uma aplicação**
- ▶ Como qualquer outra aplicação, ele **roda em uma janela**
 - O tamanho da janela é limitado pela resolução da tela
 - Número de pixels na horizontal
 - Número de pixels na vertical
 - A **resolução da tela** pode ser encontrada:

```
// captura resolução da tela (largura x altura)  
int largura = GetSystemMetrics(SM_CXSCREEN);  
int altura  = GetSystemMetrics(SM_CYSCREEN);
```

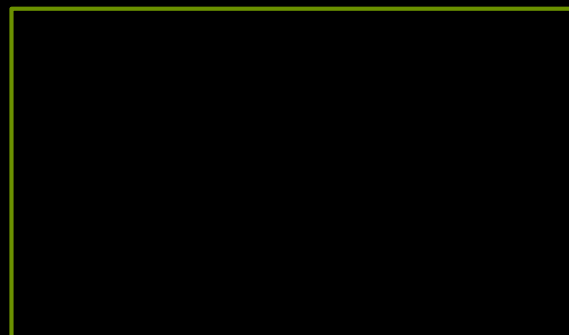


Introdução

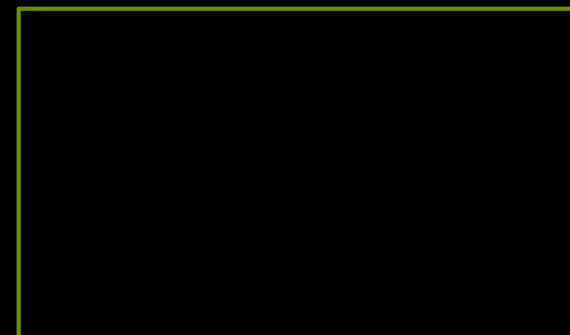
- ▶ Tipicamente os jogos rodam em um dos seguintes **modos de exibição**:
 - Modo de janela
 - Modo de tela cheia
 - Modo de janela sem borda



Em janela



Tela cheia

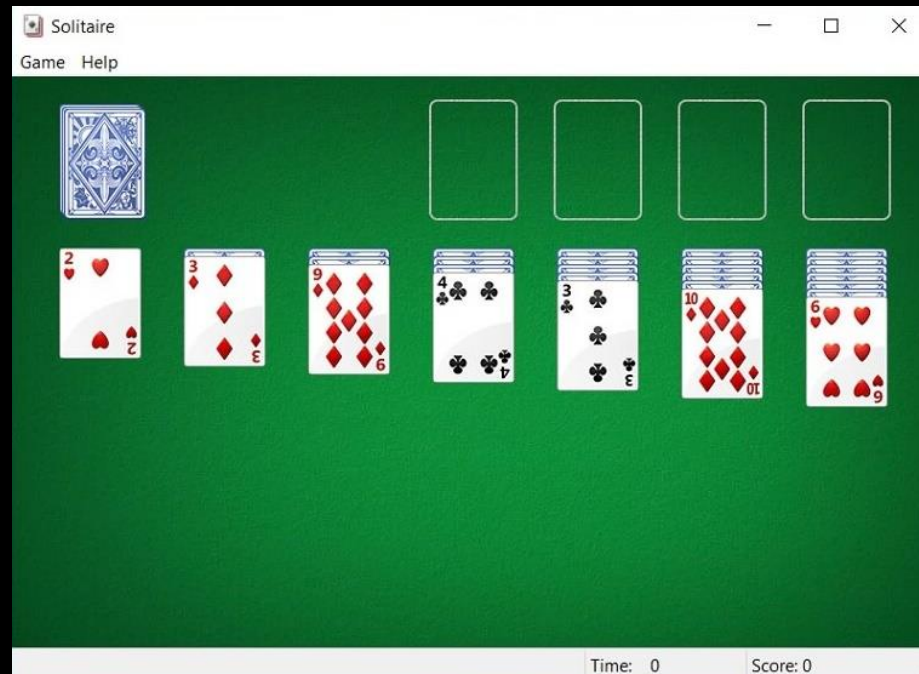


Sem borda

Modos de Exibição

► Em janela

- Ideal para jogos casuais e jogos em fase de desenvolvimento
- Área utilizável inferior a área de trabalho



Trabalhar em
janela facilita a
depuração do
jogo

Modos de Exibição

► Tela cheia

- Ambientes complexos que requerem atenção exclusiva
- O jogo tem controle total da saída de vídeo
 - Pode alterar a resolução do monitor



Dirt 3

ALT-TAB é um problema: requer a descarga e recarga da memória de vídeo

Modos de Exibição

► Janela Sem Borda

- Uma tendência mais moderna
- Funciona como uma janela mas cobre toda a tela
 - Utiliza sempre a resolução da área de trabalho



XCOM 2

Borderless Window
tem desempenho
inferior ao modo
em tela cheia*

Tela Cheia

- ▶ É obtido com uma janela “invisível”

```
HWND hwnd; // identificador da janela

hwnd = CreateWindow(
    "BasicWindow", // classe da janela
    "Aplicação", // título da janela
    WS_OVERLAPPEDWINDOW, // estilo da janela
    CW_USEDEFAULT, // posição x inicial
    CW_USEDEFAULT, // posição y inicial
    CW_USEDEFAULT, // largura inicial
    CW_USEDEFAULT, // altura inicial
    NULL, // identificador da janela pai
    NULL, // identificador do menu
    hInstance, // identificador da aplicação
    NULL); // parâmetros de criação
```

GetSystemMetrics(SM_CYSCREEN)

GetSystemMetrics(SM_CXSCREEN)

WS_POPUP

Estilo de janela sem nenhuma borda ou barra

Sem Borda

- ▶ A diferença entre o modo Tela Cheia e o modo **Janela Sem Borda** se dá na configuração do Direct3D

```
HWND hwnd; // identificador da janela

hwnd = CreateWindow(
    "BasicWindow", // classe da janela
    "Aplicação", // título da janela
    WS_POPUP, // estilo da janela
    CW_USEDEFAULT, // posição x inicial
    CW_USEDEFAULT, // posição y inicial
    GetSystemMetrics(SM_CXSCREEN), // largura inicial
    GetSystemMetrics(SM_CYSCREEN), // altura inicial
    NULL, // identificador da janela pai
    NULL, // identificador do menu
    hInstance, // identificador da aplicação
    NULL); // parâmetros de criação
```


Em Janela

► Jogo rodando em janela com tamanho fixo

```
HWND hwnd;  
  
hwnd = CreateWindow(  
    "BasicWindow",  
    "Aplicação",  
    WS_OVERLAPPEDWINDOW,  
    CW_USEDEFAULT,  
    CW_USEDEFAULT,  
    CW_USEDEFAULT,  
    CW_USEDEFAULT,  
    NULL,  
    NULL,  
    hInstance,  
    NULL);
```

// identificador da janela

// classe da janela
// título da janela
// estilo da janela
// posição x inicial
// posição y inicial
// largura inicial
// altura inicial
// identificador da janela pai
// identificador do menu
// identificador da aplicação
// parâmetros de criação

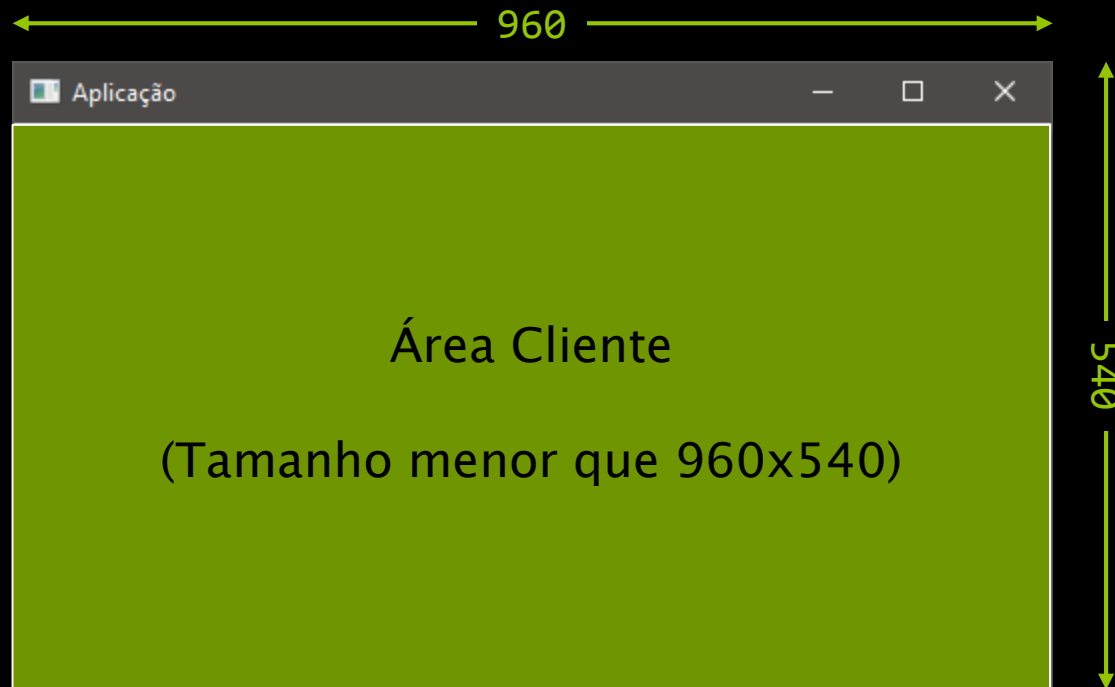
960 x 540

WS_OVERLAPPED | WS_SYSMENU

Estilo de janela sem redimensionamento

Configuração da Janela

- ▶ CreateWindow define o tamanho da janela e não da área cliente

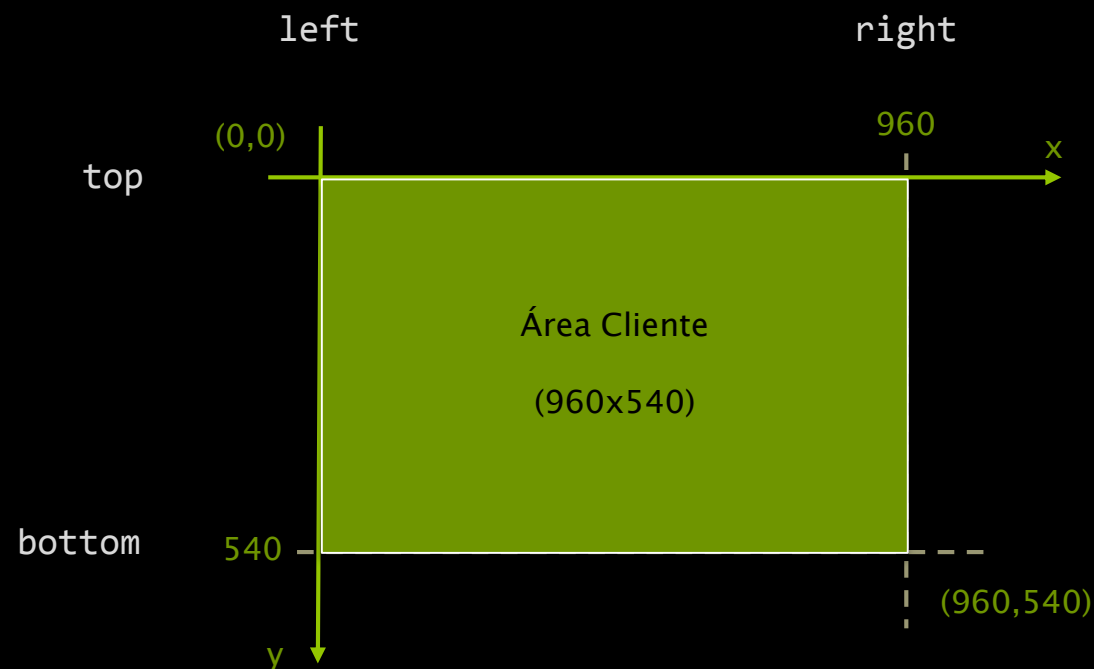


Configuração da Janela

- Podemos **ajustar o tamanho** da área cliente escrevendo o tamanho desejado em um registro **RECT**

```
// definição de RECT em windef.h
struct RECT
{
    LONG left;
    LONG top;
    LONG right;
    LONG bottom;
};

// tamanho da área cliente desejada
RECT winRect = {0, 0, 960, 540};
```



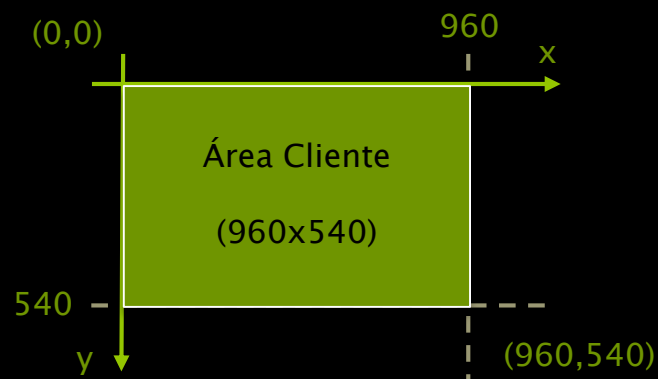
Configuração da Janela

- ▶ A função AdjustWindowRectEx permite ajustar o tamanho da área cliente

```
// retângulo com o tamanho  
// da área cliente desejada  
RECT winRect = {0, 0, 960, 540};
```

```
// ajuste do tamanho da janela  
AdjustWindowRectEx(&winRect,  
    GetWindowStyle(hwnd),  
    GetMenu(hwnd) != NULL,  
    GetWindowExStyle(hwnd));
```

```
// endereço de um registro RECT  
// estilo utilizado na janela  
// indica se a janela tem menu  
// estilos extra da janela
```



Configuração da Janela

- ▶ O registro winRect é modificado para um tamanho que comporte a área cliente desejada

```
// retângulo com o tamanho da área cliente desejada
```

```
RECT winRect = {0, 0, 960, 540};
```

```
// ajuste do tamanho da janela
```

```
AdjustWindowRectEx(&winRect,  
    GetWindowStyle(hwnd),  
    GetMenu(hwnd) != NULL,  
    GetWindowExStyle(hwnd));
```

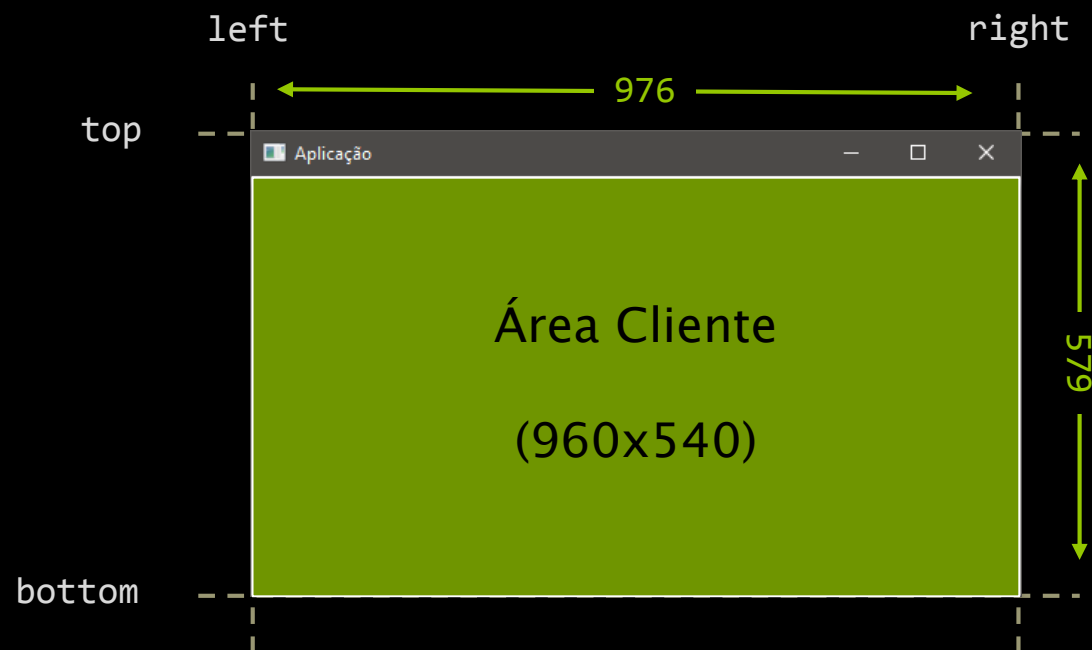
```
// após ajuste registro tem valores:
```

```
// winRect.left = -8;
```

```
// winRect.top = -31;
```

```
// winRect.right = 968;
```

```
// winRect.bottom = 548;
```



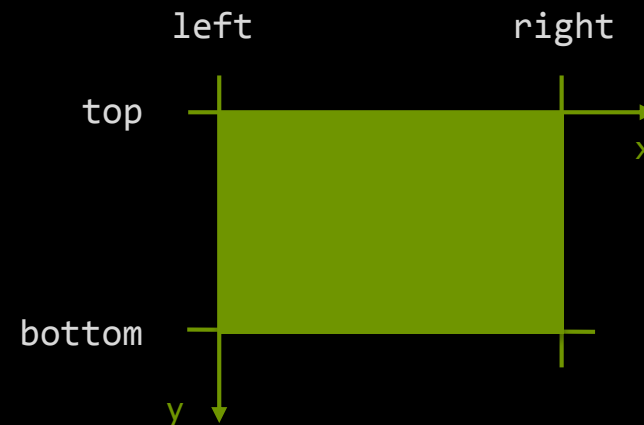
Configuração da Janela

- ▶ A função MoveWindow **atualiza o tamanho e posição da janela** com os novos valores

```
// atualiza posição da janela
int posX = (GetSystemMetrics(SM_CXSCREEN)/2) - ((winRect.right - winRect.left)/2);
int posY = (GetSystemMetrics(SM_CYSCREEN)/2) - ((winRect.bottom - winRect.top)/2);
```

```
// redimensiona janela com uma chamada a MoveWindow
```

```
MoveWindow(
    hwnd,                // id da janela
    posX,                // posição x
    posY,                // posição y
    winRect.right - winRect.left, // largura
    winRect.bottom - winRect.top, // altura
    TRUE);               // repintar
```

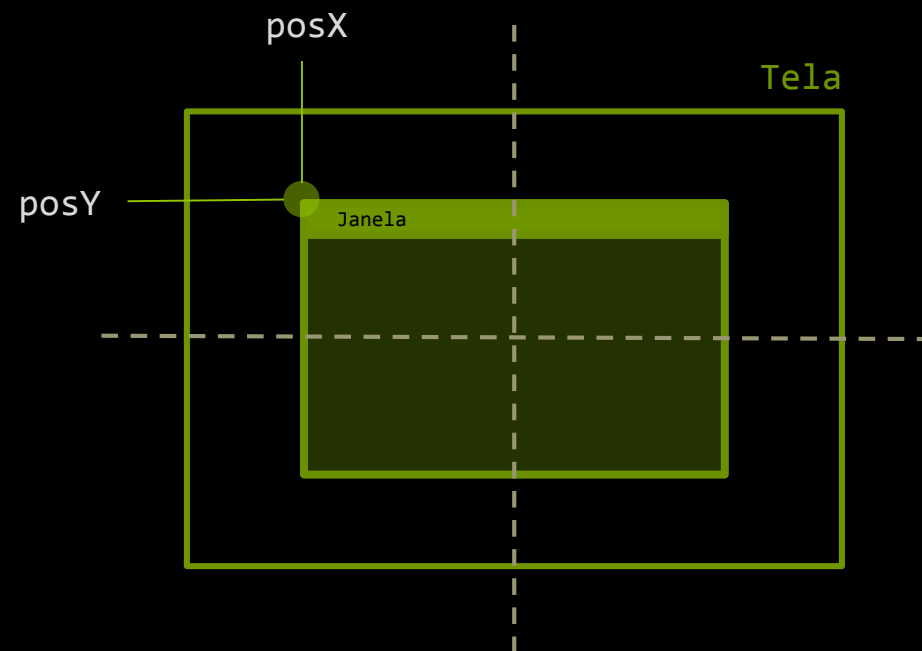


Configuração da Janela

- ▶ A chamada à função MoveWindow está também **centralizando a janela** na tela

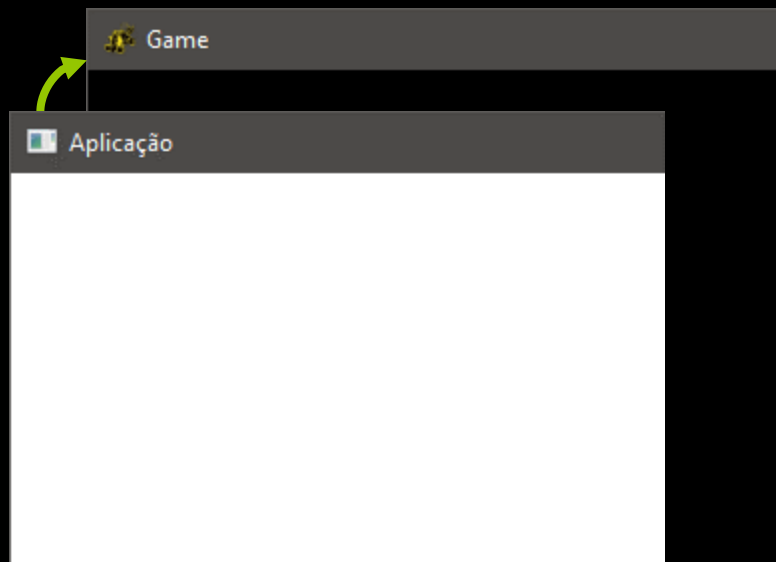
```
// atualiza posição da janela
int posX =
    (GetSystemMetrics(SM_CXSCREEN)/2) -
    ((winRect.right - winRect.left)/2);

int posY =
    (GetSystemMetrics(SM_CYSCREEN)/2) -
    ((winRect.bottom - winRect.top)/2);
```

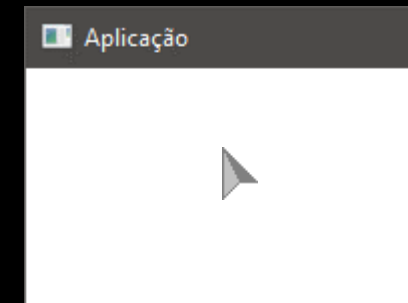


Ícones e Cursores

- ▶ O ícone e o cursor de uma aplicação podem ser personalizados



Janela



Cursor



Barra de Tarefas




Ícones e Cursores

- ▶ A macro **MAKEINTRESOURCE** deve ser usada para carregar um recurso personalizado

```
wndclass.hIcon = LoadIcon(hInstance, MAKEINTRESOURCE(IDI_ICON));  
wndclass.hCursor = LoadCursor(hInstance, MAKEINTRESOURCE(IDC_CURSOR));
```

- ▶ O recurso deve ser declarado em um arquivo .rc, tipicamente chamado de **Resources.rc**

```
#include "Resources.h"
```

IDI_ICON	ICON	"Resources\\Icon.ico"
IDC_CURSOR	CURSOR	"Resources\\Cursor.cur"
		
Id	Tipo	Localização

O id é uma constante simbólica (numérica) que normalmente é declarada dentro de um arquivo "Resources.h".

Resumo

- ▶ Um jogo pode ter sua **exibição**:
 - Em tela cheia ou em janela
- ▶ **CreateWindow** define o tamanho da janela
 - As funções **AdjustWindowRectEx** e **MoveWindow** podem ser usadas para definir o tamanho da área cliente
- ▶ Ícones e cursores personalizados podem ser usados:
 - Através da macro **MAKEINTRESOURCE**
 - Com a criação de um arquivo Resources.rc