

# Programação de Interface Visual

Alexandre Abdalla

GDP

DCC - UFRJ

# Conteúdo

- Ø Programação de Interface Visual com a GUI do Allegro
- Ø A.G.U.P. (Allegro GUI Un-uglyfication Project)

# Introdução

- ∅ Simula a P.O.O. em C
- ∅ Uso constante de Ponteiros para funções
- ∅ Estrutura (Classe) padrão: DIALOG e MENU
- ∅ Muitas Constantes e Pré-definições

# Inicialização do Agup

Ø void agup\_init(AL\_CONST AGUP\_THEME \*);

Ø void agup\_shutdown(void);

extern AL\_CONST AGUP\_THEME \*aalg\_theme;

extern AL\_CONST AGUP\_THEME \*aase\_theme;

extern AL\_CONST AGUP\_THEME \*abeos\_theme;

extern AL\_CONST AGUP\_THEME \*agtk\_theme;

extern AL\_CONST AGUP\_THEME \*ans\_theme;

extern AL\_CONST AGUP\_THEME \*aphoton\_theme;

extern AL\_CONST AGUP\_THEME \*awin95\_theme;

# Estruturas Iniciais

- ∅ `extern int gui_bg_color; (agup_bg_color)`
- ∅ `extern int gui_mg_color; (agup_mg_color)`
- ∅ `extern int gui_fg_color; (agup_fg_color)`
- ∅ `enable_hardware_cursor(void);`
- ∅ `set_window_title(const char *name);`

# Estruturas Iniciais

Ø `set_close_button_callback(void (*proc)(void));`

```
volatile int close_button_pressed = FALSE;
void close_button_handler(void)
{
    close_button_pressed = TRUE;
}
END_OF_FUNCTION(close_button_handler)
...
allegro_init();
LOCK_FUNCTION(close_button_handler);
set_close_button_callback(close_button_handler);
...

while (!close_button_pressed) do_stuff();
```

# Janelas de Alerta

```
Ø int alert(const char *s1, *s2, *s3, const char *b1, *b2, int c1, c2);  
n BITMAP *img = load_bitmap(img_path, NULL);  
  if (!img)  
    alert(img_path, "not found.", NULL,  
          "&Continue", NULL, 'c', 0);
```

# Janelas de Alerta

```
Ø int alert3(const char *s1, *s2, *s3, const char *b1, *b2, *b3, int c1, c2, c3);  
switch (alert3("Vc quer frango ?", "Peixe ?", "Ou Porco ?", "&Frango", "P&eixe",  
    "P&orco", 'f', 'e', 'o'))  
{  
    case(1):  
        ...  
    break;  
    case(2):  
        ...  
    break;  
    case(3):  
        ...  
    break;  
}
```



# Estrutura DIALOG

- ∅ Estrutura chave da GUI
- ∅ Geralmente em forma de vetor.
- ∅ Define qual será a componente GUI a ser usada

# Estrutura DIALOG

## Ø Elementos:

- Ø `int (*proc)(int, DIALOG *, int);`
- Ø `int x, y, w, h;`
- Ø `int fg, bg;`
- Ø `int key;`
- Ø `int flags;`
- Ø `int d1, d2;`
- Ø `void *dp, *dp2, *dp3;`

## Ø Elementos:

Ø `int (*proc)(int, DIALOG *, int);`

Ø `int x, y, w, h;`

Ø `int fg, bg;`

Ø `int key;`

Ø `int flags;`

Ø `int d1, d2;`

Ø `void *dp, *dp2, *dp3;`

# Componentes GUI - Agup

```
n int d_agup_box_proc(int, DIALOG *, int);
n int d_agup_shadow_box_proc(int, DIALOG *, int);
n int d_agup_button_proc(int, DIALOG *, int);
n int d_agup_push_proc(int, DIALOG *, int);
n int d_agup_check_proc(int, DIALOG *, int);
n int d_agup_radio_proc(int, DIALOG *, int);
n int d_agup_icon_proc(int, DIALOG *, int);
n int d_agup_edit_proc(int, DIALOG *, int);
n int d_agup_list_proc(int, DIALOG *, int);
n int d_agup_text_list_proc(int, DIALOG *, int);
n int d_agup_textbox_proc(int, DIALOG *, int);
n int d_agup_slider_proc(int, DIALOG *, int);
n int d_agup_menu_proc(int, DIALOG *, int);
n int d_agup_window_proc(int, DIALOG *, int);
n int d_agup_clear_proc(int, DIALOG *, int);
n int d_agup_text_proc(int, DIALOG *, int);
n int d_agup_ctext_proc(int, DIALOG *, int);
n int d_agup_rtext_proc(int, DIALOG *, int);
```

# Retorno das Componentes GUI

- Ø D\_O\_K

  - Ø Retorno Padrão

- Ø D\_CLOSE

  - Ø Mensagem de fechamento de dialogo

- Ø ...

## Ø Elementos:

Ø int (\*proc)(int, DIALOG \*, int);

Ø int x, y, w, h;

Ø int fg, bg;

Ø int key;

Ø int flags;

Ø int d1, d2;

Ø void \*dp, \*dp2, \*dp3;

# Flags

- Ø D\_EXIT

- Ø Fecha o Dialog quando selecionado

- Ø D\_SELECTED

- Ø Informa que esta selecionado

- Ø D\_DISABLED

- Ø Desabilita a componente

- Ø ...

# Estrutura MENU

- Ø Menu superior padrão de aplicativos com GUI
- Ø Geralmente em forma de vetor
- Ø Permite MENUs aninhados

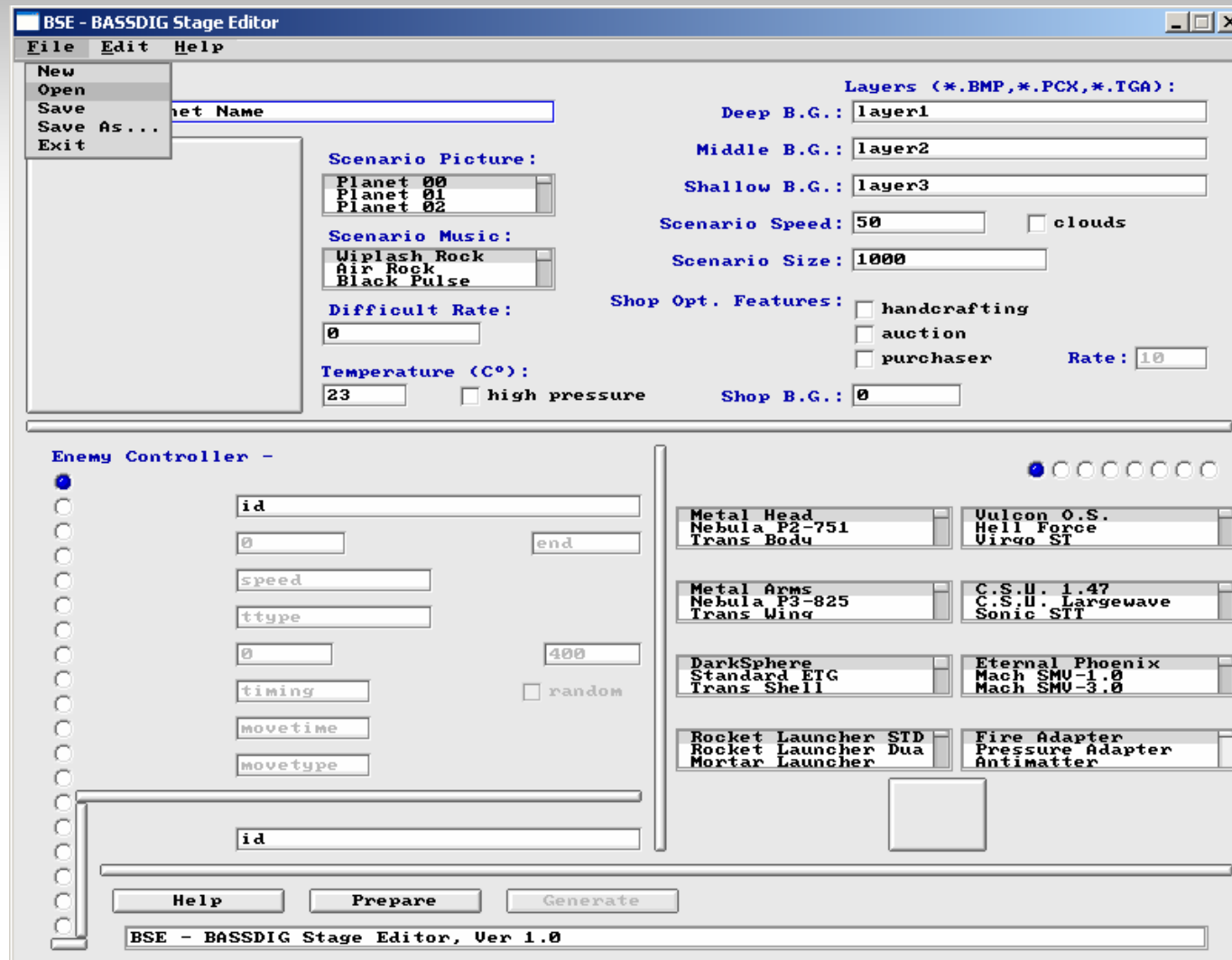


# Estrutura MENU

## Ø Elementos

- Ø char \*text;
- Ø int (\*proc)(void);
- Ø struct MENU \*child;
- Ø int flags;
- Ø void \*dp;

# Exemplo



# Perguntas ?