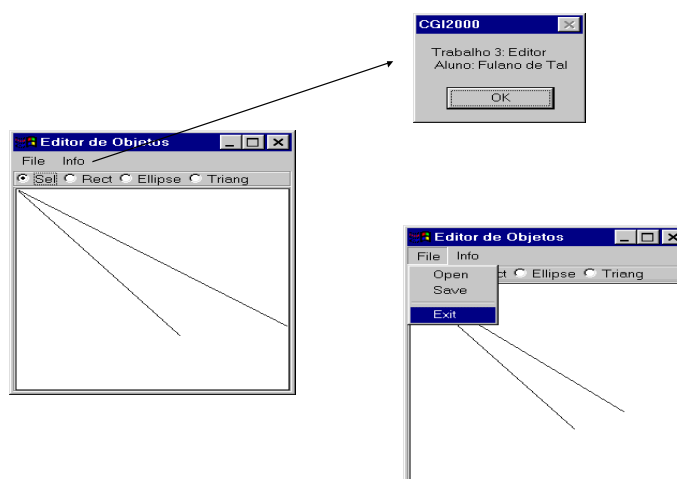
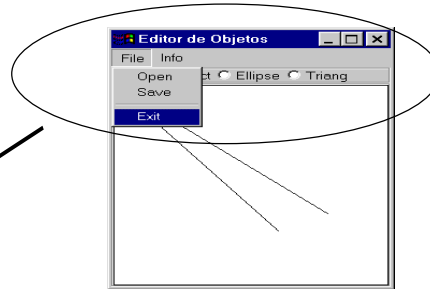


## T3 Editor

### Dialogo Principal



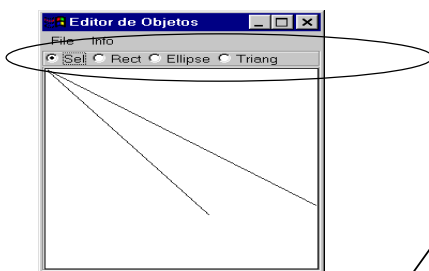
## Criando o Menu de Barra



```
/* Criando o menu de arquivos */
Ihandle * open_item = IupItem( "Open", "open_act");
Ihandle * save_item = IupItem( "Save", "save_act");
Ihandle * exit_item = IupItem( "Exit", "exit_act");
Ihandle * arq_menu=IupMenu(open_item, save_item,IupSeparator(), exit_item, NULL);

/* Criando o menu de barra */
Ihandle * arq_submenu = IupSubmenu("File", arq_menu);
Ihandle * info_item  = IupItem( "Info", "info_act");
Ihandle * main_menu  = IupMenu(arq_submenu, info_item, NULL);
```

## Criando um Radio Button

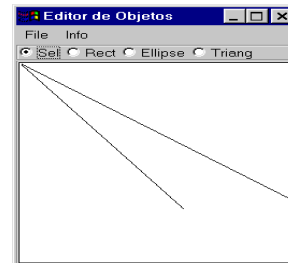


```
/* Radio que contem a selecao do modo de interacao com usuario */
Ihandle * sel_tog  = IupToggle ( "Sel",    "sel_act");
Ihandle * rect_tog = IupToggle ( "Rect",   "rect_act");
Ihandle * ellipse_tog = IupToggle ( "Ellipse", "ellipse_act");
Ihandle * triang_tog = IupToggle ( "Triang", "triang_act");

Ihandle * sel_hbox=IupHbox(sel_tog,rect_tog,ellipse_tog,triang_tog,IupFill(),NULL);
Ihandle * mode_rad = IupRadio( IupFrame(sel_hbox) );
```

## Criando o Canvas e o Diálogo

Criando o Canvas e o Diálogo com o Rádio e o Canvas.

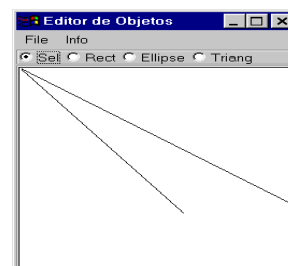


```
/* Criando o canvas (CD) */
Ihandle * iup_canvas = IupCanvas("repaint_act");

/* Dialogo principal contendo os botoes, o menu e o Canvas */
Ihandle * main_vbox = IupVbox(mode_rad, IupFrame(iup_canvas),NULL);
Ihandle * main_dlg = IupDialog(main_vbox);

/* Atributos do Canvas */
IupSetAttribute(iup_canvas,IUP_RESIZE_CB,"resize_act");
IupSetAttribute(iup_canvas,IUP_BUTTON_CB,"button_act");
IupSetAttribute(iup_canvas,IUP_MOTION_CB,"motion_act");
IupSetAttribute(iup_canvas,IUP_RASTERSIZE,"256x256");
```

## Acrescentando o Menu e o Título



```
/* Atributos do dialogo */
IupSetAttribute(main_dlg, IUP_TITLE, "Editor de Objetos");
IupSetHandle("main_menu_handle", main_menu);
IupSetAttribute(main_dlg, IUP_MENU, "main_menu_handle");
```

```
/* exhibe o dialogo */
IupShow(main_dlg);
```

EXIBE

```
canvasCreate(iup_canvas);
```

Liga com o  
Sistema Gráfico

## Programa Principal

```
void main(void){
    IupOpen();

    initInterface();

    /* Registrando a Callbacks */
    IupSetFunction("open_act", (Icallback) open_cb);
    IupSetFunction("save_act", (Icallback) save_cb);
    ...

    IupMainLoop();
    IupClose();
}
```

## Callbacks

```
int info_cb(void)
{
    IupMessage("CGI2000","Trabalho 3: Editor \n Aluno: Fulano de Tal");

    /* Retornando para o "iup main loop" */
    return IUP_DEFAULT;
}
```

```
int motion_cb(Ihandle* iupcanvas, int x, int y)
{
    /* Tratando o evento */
    printf("<%d,%d>",x,y);
    canvasDrawTempLine(iupcanvas,0,0,x,y);

    /* Retornando para o "iup main loop" */
    return IUP_DEFAULT;
}
```

## Contexto de Desenho no Canvas

```
/*- Contexto deste canvas (permite aplicação com vários canvas):-----*/
typedef struct _canvas_context {
    cdCanvas *cd_canvas;
    struct _tmp_line {
        int flag;
        int x0,y0,x1,y1;
    } tline;
} CanvasContext;

/* cria um contexto para desenhar neste canvas */
void canvasCreate(Ihandle *iup_canvas) {
    CanvasContext *cc = (CanvasContext *) malloc(sizeof(CanvasContext));
    cc->cd_canvas = cdCreateCanvas(CD_IUP, iup_canvas );
    cc->tline.flag =0;
    IupSetAttribute(iup_canvas,"CanvasContext",(Ihandle *)cc);
    return;
}
```

## Contexto de Desenho no Canvas

```
/* desenha uma linha temporaria (que vai ser apagada depois) */
void canvasDrawTempLine(Ihandle *iup_canvas, int x0, int y0, int x1, int y1) {
    CanvasContext *cc=(CanvasContext *)IupGetAttribute(iup_canvas,"CanvasContext");
    if (cc==NULL) return;

    cdActivate(cc->cd_canvas);
    cdWriteMode(CD_NOT_XOR);

    /* apaga a linha anterior (se existir) */
    if (cc->tline.flag!=0)
        cdLine(cc->tline.x0,cc->tline.y0,cc->tline.x1,cc->tline.y1);

    /* desenha a nova linha */
    cdUpdateYAxis(&y0); /* troca de sistema do IUP para o do CD */
    cdUpdateYAxis(&y1); /* troca de sistema do IUP para o do CD */
    cdLine(x0,y0,x1,y1);

    /* salva a linha para ser apagada */
    cc->tline.flag=1;
    cc->tline.x0=x0;
    cc->tline.y0=y0;
    cc->tline.x1=x1;
    cc->tline.y1=y1;
}
```