

CUADERNO DE EJERCICIOS Y PRACTICAS C++ PROGRAMACION II

Contenido

Ejercicio 00a – Ventana de mensaje	4
Ejercicio 00b - Ventana	4
Ejercicio 01 – Forma Simple.....	6
Ejercicio 02 - Botones.....	7
Ejercicio 03 - TextBox.....	8
Ejercicio 04 – Argumentos de linea comandos.....	10
Ejercicio 05 – Aplicación de consola	11
Ejercicio 06 - ColorButton	12
Ejercicio 07 – Eventos de Forma.....	13
Ejercicio 08 – Funciones del sistema: Apagado	14
Ejercicio 09 - Mouse.....	15
PRACTICA 1	17
PRACTICA 2	17
PRACTICA 3	18
PRACTICA 4	18
PRACTICA 5	19
Ejercicio 10 - MenuPopUp	19
Ejercicio 11 - MDIForm	21
Ejercicio 12 - Timers.....	21
Ejercicio 13 - Video.....	23
Ejercicio 14 - TableView	23
Ejercicio 15 - ScrollBars	24
Ejercicio 16 – ImageBox	25
Ejercicio 17 – ImageBox desde EXE.....	26
Ejercicio 18 – ImageBox desde Internet	27
Ejercicio 19 – Información del Sistema	27
PRACTICA 6	29
PRACTICA 7	29
PRACTICA 8	30
PRACTICA 9	31
PRACTICA 10	31
Ejercicio 20 - DragNDrop.....	32
Ejercicio 21 – Tip del Dia	33
Ejercicio 22 - TipOfDay.....	34
Ejercicio 23 - Tab	35
Ejercicio 24 – Escribir archivos Excel.....	37
Ejercicio 25 – System Tray	38
Ejercicio 27 – Subclase ListBox.....	40
Ejercicio 28 – Splitter Vertical	41
Ejercicio 29 – Splitter Horizontal.....	43

Ejercicio 30 - Menus.....	44
PRACTICA 11	45
PRACTICA 12	46
Ejercicio 31 - MiniNotepad	47
Ejercicio 32 - Formas.....	49
Ejercicio 33 - ListBox	50
Ejercicio 34 - ComboBox	51
Ejercicio 35 – ListBox MultiColumna.....	53
Ejercicio 36 – ComboBox con Iconos	54
Ejercicio 37 - ListView	56
Ejercicio 38 – Icono de Forma	57
Ejercicio 39 – Slider Horizontal	58
Ejercicio 40 – Slider Vertical.....	60
Ejercicio 41 - CheckBox	61
Ejercicio 42 - RadioButtons	62
Ejercicio 43 - ProgressBar	63
Ejemplo 44 - StatusBar.....	64
PRACTICA 13	66
PRACTICA 14	68
PRACTICA 15	70
PRACTICA 16	70
PRACTICA 17	71
Ejercicio 45 – ScrollBars 2	71
Ejercicio 46 - ToolBar	72
Ejercicio 47 - Treeview	74
Ejercicio 48 – RichTextBox Intermedio	76
Ejercicio 49 – Arrays Asociativos.....	78
Ejercicio 50 – Grupos de Objetos.....	79
Ejercicio 51 - CoolBars.....	81
Ejercicio 52 – Toolbar y CoolBar	83
PRACTICA 18	85
PRACTICA 19	85
PRACTICA 20	85

Ejercicio 00a – Ventana de mensaje

```
#include <windows.h>
int WINAPI WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance,
    LPSTR lpCmdLine, int nCmdShow)
{
    MessageBox(NULL, "Goodbye, cruel world!", "Note", MB_OK);
    return 0;
}
```

Ejercicio 00b - Ventana

```
#include <windows.h>

/* Declaración del procedimiento de ventana */
LRESULT CALLBACK WindowProcedure (HWND, UINT, WPARAM, LPARAM);

int WINAPI WinMain (HINSTANCE hThisInstance,
    HINSTANCE hPrevInstance,
    LPSTR lpzArgument,
    int nFunsterStil)
{
    HWND hwnd;          /* Manipulador de ventana */
    MSG mensaje;         /* Mensajes recibidos por la aplicación */
    WNDCLASSEX wincl;    /* Estructura de datos para la clase de ventana */

    /* Estructura de la ventana */
    wincl.hInstance = hThisInstance;
    wincl.lpszClassName = "NUESTRA_CLASE";
    wincl.lpfnWndProc = WindowProcedure; /* Esta función es invocada por Windows */
    wincl.style = CS_DBLCLKS;           /* Captura los doble-clicks */
    wincl.cbSize = sizeof (WNDCLASSEX);

    /* Usar icono y puntero por defecto */
    wincl.hIcon = LoadIcon (NULL, IDI_APPLICATION);
    wincl.hIconSm = LoadIcon (NULL, IDI_APPLICATION);
    wincl.hCursor = LoadCursor (NULL, IDC_ARROW);
    wincl.lpszMenuName = NULL;          /* Sin menú */
    wincl.cbClsExtra = 0;               /* Sin información adicional para la */
    wincl.cbWndExtra = 0;               /* clase o la ventana */
    /* Usar el color de fondo por defecto para la ventana */
}
```

```
wincl.hbrBackground = GetSysColorBrush(COLOR_BACKGROUND);

/* Registrar la clase de ventana, si falla, salir del programa */
if(!RegisterClassEx(&wincl)) return 0;

/* La clase está registrada, crear la ventana */
hwnd = CreateWindowEx(
    0,          /* Posibilidades de variación */
    "NUESTRA_CLASE", /* Nombre de la clase */
    "Ejemplo 001", /* Texto del título */
    WS_OVERLAPPEDWINDOW, /* Tipo por defecto */
    CW_USEDEFAULT, /* Windows decide la posición */
    CW_USEDEFAULT, /* donde se coloca la ventana */
    544,          /* Ancho */
    375,          /* Alto en pixels */
    HWND_DESKTOP, /* La ventana es hija del escritorio */
    NULL,         /* Sin menú */
    hThisInstance, /* Manipulador de instancia */
    NULL          /* No hay datos de creación de ventana */
);

/* Mostrar la ventana */
ShowWindow(hwnd, SW_SHOWDEFAULT);

/* Bucle de mensajes, se ejecuta hasta que haya error o GetMessage devuelva FALSE */
while(TRUE == GetMessage(&mensaje, NULL, 0, 0))
{
    /* Traducir mensajes de teclas virtuales a mensajes de caracteres */
    TranslateMessage(&mensaje);
    /* Enviar mensaje al procedimiento de ventana */
    DispatchMessage(&mensaje);
}

/* Salir con valor de retorno */
return mensaje.wParam;
}

/* Esta función es invocada por la función DispatchMessage() */
LRESULT CALLBACK WindowProcedure (HWND hwnd, UINT mensaje, WPARAM wParam, LPARAM lParam)
{
    switch (mensaje)          /* manipulador de mensaje */
    {
        case WM_DESTROY:
```

```
PostQuitMessage (0);    /* Envía el mensaje WM_QUIT a la cola de mensajes */
break;
default:                /* Mensajes que no queremos manejar */
    return DefWindowProc (hwnd, mensaje, wParam, lParam);
}

return 0;
}
```

Ejercicio 01 – Forma Simple

```
/* ej01_FormaSimple
Ejemplo de creacion de una ventana usando RAD C++ */
#include <radc++.h>

Form form1("Ventana Simplee - RAD C++ Ejemplo");

rad_main()

    //main program code goes here

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es la librería principal que se incluye?	
Explica la sintaxis del comando Form	
¿Cuál es la funcion principal de este programa?	

Ejercicio 02 - Botones

```
/* ej02_BotonesYEventosForma
Ejemplo para crear una forma, agregarle un boton y
ver alguno de los eventos que puede tener dicha forma */
#include <radc++.h>

//create new form
Form Form1("Evento Boton - RAD C++ Ejemplo");

//create new button
Button Button1("Soy un boton", AUTO_ID, 10, 15, 100, 150, Form1);

//create new control procedure for form, procedure1 is procedure name
FormProcedure procedure1 (FormProcArgs) {

    //capture event ON_CLOSE which is fired when form is closed, [X] is clicked
    ON_CLOSE() {
        //exit application
        Application.close();
    }
    //capture button click, means Button has been clicked
    ON_COMMAND_BY ( Button1 ) {
        Form1.msgBox("Hola Universo!");
    }

    //always return 0
    return 0;
}

//rad c++ main program
rad_main()

    //attach the created procedure with form
    Form1.procedure = procedure1;

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
Explica la sintaxis del control Button	
¿Qué hace FormProcedure?	
¿Cuáles son los eventos de esta forma?	
¿Cuál es el comando que muestra una caja de dialogo de mensaje?	

Ejercicio 03 - TextBox

```
/* ej03_EventosTexto
```

```
En este ejemplo veremos como se crea una forma, como se pone texto
dentro de la forma con un control de texto y como se usan algunos
eventos de dicho control */
```

```
#include <radc++.h>
```

```
//create new form
```

```
Form Form1("Eventos Texto - RAD C++ Ejemplo");
```

```
//create new text box
```

```
TextBox text1("Teclee aqui", AUTO_ID, 10, 15, 200, 25, Form1);
```

```
//create new control procedure for form, procedure1 is procedure name
```

```
FormProcedure procedure1 (FormProcArgs) {
```

```
    //capture event ON_CLOSE which is fired when form is closed, [X] is clicked
```

```
    ON_CLOSE() {
```

```
        //exit application
```

```
        Application.close();
```

```
    }
```

```
    //capture text change event, means text has been changed
```

```
    ON_TEXT_CHANGED ( text1 ) {
```

```
        Form1.caption = "Cambio a : " + text1.text;
```

```
    }
```

```
    //always return 0
```

```
    return 0;
```

```
}

//rad c++ main program
rad_main()

    //attach the created procedure with form
    Form1.procedure = procedure1;

    //set focus to textbox when program starts
    text1.focus();

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
Explica la sintaxis del control TextBox	
¿Qué hace Application.close()?	
¿Qué hace el evento ON_TEXT_CHANGED()?	
¿Qué hace la propiedad caption?	

Ejercicio 04 – Argumentos de linea comandos

```
/* ej04_ArgumentosCmd
```

```
En este ejemplo veremos como pasar argumentos desde la linea  
de comando a la forma que estamos creando */
```

```
#include <radc++.h>
```

```
Form form1("Argumentos Linea de Comando - RAD C++ Ejemplo");
```

```
//application argyments must be separated by tab key (\t char 9) not space
```

```
rad_main()
```

```
    int totalArguments = Application.arguments.length;
```

```
    String arg1Value = Application.arguments[0];
```

```
    form1.msgBox(arg1Value, "Total "+str(totalArguments) + " argumentos");
```

```
rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es lo que hace el programa?	

Ejercicio 05 – Aplicación de consola

/* ej05_AplicacionConsola

En este ejemplo usamos la ventana de consola
para capturar una variable de cadena, la leemos
y luego escribimos su valor */

#include <conio.h>

Console con(true);

int main() {

con.write("Introduzca una cadena:");

String value = con.read();

con.write("Ha introducido : "+value + "\n\nPresione Enter para continuar..");

con.read();

return 0;

}

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué hace el programa?	
¿Qué hace con.write()?	
¿Qué hace con.read()?	
¿Qué es el tipo de dato STRING?	

Ejercicio 06 - ColorButton

```
/* ej06_ColorBoton
```

```
Este ejemplo es para mostrar como cambiar el color a  
un control de boton, asi como usar los controles de label  
caja numerica y algunos eventos */
```

```
#include <radc++.h>
```

```
Form          form1("Color de boton - RAD C++ Ejemplo");  
ColorButton col_btn(AUTO_ID,100,60,200,50,form1,0xD17878);  
//0xD17878 is default color in hex which equals to = 13727864
```

```
Label  label1("Cambie padding:",-1,100,130,100,20,form1);  
NumberBox  pad_txt("5",AUTO_ID,100,150,50,25,form1);
```

```
Label  label2("Cambie Color:",-1,200,130,100,20,form1);  
NumberBox  col_txt("13727864",AUTO_ID,200,150,100,25,form1);
```

```
FormProcedure form1Proc(FormProcArgs) {  
    ON_CLOSE()  Application.close(); //close application when form is closed  
  
    ON_COMMAND_BY(col_btn) { //button clicked  
        COLORREF last_color = col_btn.color;  
        if(form1.selectColor(last_color)) {  
            col_btn.color = last_color;  
            col_txt.text = str(last_color);  
        }  
    }  
  
    ON_TEXT_CHANGED(pad_txt) { //user entered new padding value  
        col_btn.padding = val(pad_txt.text);  
    }  
  
    ON_TEXT_CHANGED(col_txt) { //user entered new color value  
        col_btn.color = val(col_txt.text);  
    }  
  
    return 0;  
}
```

```
rad_main()
```

```
form1.procedure = form1Proc;
pad_txt.setLimit(2); //limit number of digits for numberbox pad_txt
```

```
rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué controles usamos en este programa?	
¿Qué hace Label?	
¿Cuál es la diferencia de NumberBox con TextBox?	
¿Qué hace la funcion STR()?	
¿Qué hace la funcion VAL()?	

Ejercicio 07 – Eventos de Forma

```
/* ej07_EventosForma
Ejemplo para usar los eventos que contiene una forma
usando RadC++ */
#include <radc++.h>

//create new form
Form Form1("Eventos Forma - RAD C++ Ejemplo");

//create new procedure for form, procedure1 is procedure name
FormProcedure procedure1 (FormProcArgs) {

    //capture event ON_CLOSE which is fired when form is closed, [X] is clicked
    ON_CLOSE() {
        //show simple message box
        Form1.msgBox("Estoy saliendo");
        //exit application
        Application.close();
    }

    //always return 0
    return 0;
}
```

```
//rad c++ main program
rad_main()

    //attach the created procedure with form
    Form1.procedure = procedure1;

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué hace el programa?	
¿Qué eventos se ejecutan en esta forma?	

Ejercicio 08 – Funciones del sistema: Apagado

```
/* ej08_Apagado
Ejemplo que muestra algunas funciones del sistema
desde C++ */
#include <radc++.h>

Form form1("Reset/ShutDown/LogOff - RAD C++ Ejemplo");
Button btnL("Log Off", AUTO_ID,35,100,100,30,form1);
Button btnS("Shut Down",AUTO_ID,145,100,100,30,form1);
Button btnR("Restart", AUTO_ID,255,100,100,30,form1);

FormProcedure form1Proc(FormProcArgs) {

    ON_CLOSE() Application.close();

    ON_COMMAND_BY(btnL) System.logOff();

    ON_COMMAND_BY(btnS) System.shutDown();

    ON_COMMAND_BY(btnR) System.restart();

    return 0;
}
```

```
rad_main()

        form1.procedure = form1Proc;

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué hace el programa?	
¿Qué hace Application.close()?	
¿Qué hace System.logOff()?	
¿Qué hace System.shutdown()?	
¿Qué hace System.restart()?	

Ejercicio 09 - Mouse

```
/* ej09_Mouse
Este ejemplo muestra algunas operaciones con el mouse
y como controlarlo */
#include <radc++.h>

Form form1("Operaciones con Raton - RAD C++ Ejemplo");

TextBox t1("X",AUTO_ID,100,100,50,30,form1);
TextBox t2("Y",AUTO_ID,160,100,50,30,form1);

Button b1("&Esconder Mouse",AUTO_ID,100,150,150,30,form1);
Button b2("&Mostrar Mouse",AUTO_ID,100,210,150,30,form1);

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_TEXT_CHANGED(t1)
```

```
        Mouse.x = val(t1.text); //set mouse x coordinates

    ON_TEXT_CHANGED(t2)
        Mouse.y = val(t2.text); //set mouse y coordinates

    ON_COMMAND_BY(b1)
        Mouse.visible=false; //hide mouse cursor

    ON_COMMAND_BY(b2)
        Mouse.visible=true; //show mouse cursor if not visible

    return 0;
}

//Where Mouse is a global object of class _RAD_MOUSE

rad_main()

    form1.procedure = proc;

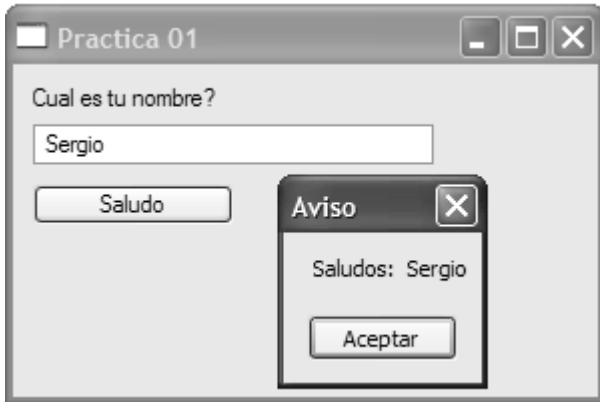
rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué hace el programa?	
¿Cómo escondemos el cursor del mouse?	

PRACTICA 1

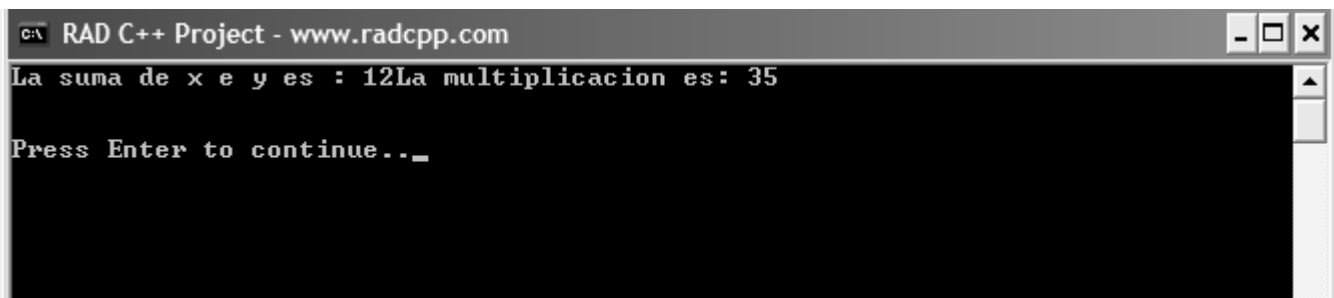
Crear un programa que pida el nombre del usuario y luego muestre un saludo con el nombre introducido en una ventana de dialogo de mensaje (o de forma)



PRACTICA 2

Realiza el ejercicio 2 del Cuaderno de Ejercicios de Programacion I: C++ con la librería RAD usando el metodo de consola.

Si no recuerdas este ejercicio, es sobre los tipos de datos de C++. Pídele a tu profesor que te asesore.



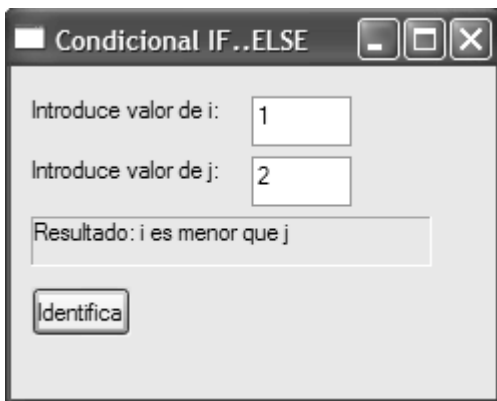
PRACTICA 3

Realiza la misma practica anterior, pero ahora usando los controles GUI para hacerlo en una interfaz grafica.



PRACTICA 4

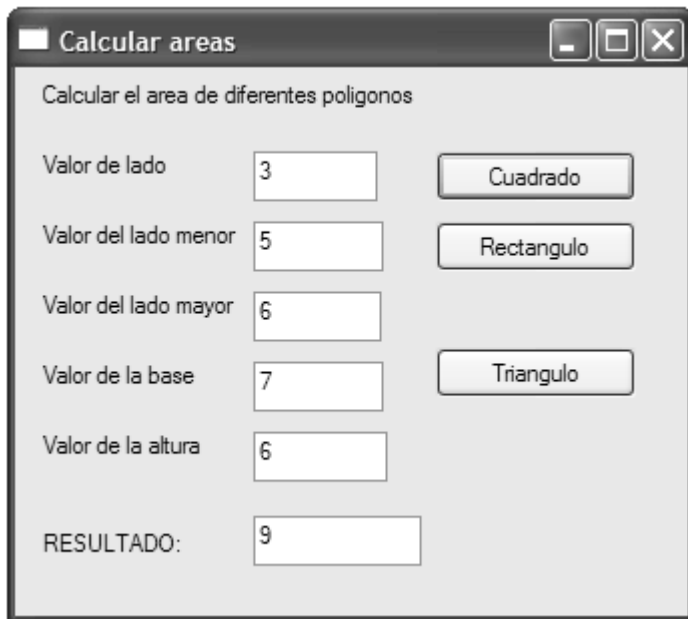
Realiza el ejercicio 11 del Cuaderno de Ejercicios de Programacion I con la librería RAD usando los controles. Como recordatorio, este ejercicio se basa en la sentencia IF...ELSE para comparar dos variables numericas. Era para practicas la sentencia IF...ELSE, sin embargo, ahora debes pedirle al usuario que introduzca los valores para hacer la comparacion.



PRACTICA 5

Realiza el ejercicio 16 del Cuaderno de Ejercicios de Programacion I con la librería RAD.

Como recordatorio, este ejercicio se basa en la sentencia SWITCH para decision. Pero en el entorno grafico no lo necesitaremos.



Ejercicio 10 - MenuPopUp

/* ej10_MenuPopUp

Este ejemplo muestra como abrir un submenu dentro de una forma */

```
#include <radc++.h>
```

```
Form form1("Popup menu - RAD C++ Example");
```

```
PopupMenu popup;
```

```
MenuItem i1,i2;
```

```
FormProcedure form1Proc(FormProcArgs) {  
    ON_CLOSE() Application.close();
```

```

    ON_RIGHT_CLICK() {
        popup.show(form1); //show at form 1
    }

    ON_COMMAND_BY(i1) form1.text = "i1";
    ON_COMMAND_BY(i2) form1.text = "i2";

    return 0;
}

rad_main()

    form1.procedure = form1Proc;

    i1 = popup.add("I am item 1",AUTO_ID);
    i2 = popup.add("I am item 2",AUTO_ID);
        popup.addSeparator();

    Menuitem n = popup.addPopup("oooh a popup menu"); //sub-menus not tracked
        n.add("Sub 1",AUTO_ID);
        n.add("Sub 2",AUTO_ID);
        n.add("Sub 3",AUTO_ID);

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué hace el evento ON_RIGHT_CLICK()?	
¿Cómo creamos un menú Popup?	
¿Cómo agregamos una opción al menú PopUp?	
¿Qué es AUTO_ID?	

Ejercicio 11 - MDIForm

```

/* ej11_MDIForm
Ejemplo de como crear ventanas dentro de otras ventanas
es lo que llamamos Forma MDI */

#include <radc++.h>

//Create new MDI Form with title "MDI Form"
MDIForm form1("MDI Form");

//Create new child form within 'form1' by calling its function 'createChild'
//NOTE:newly created child form will be accessible through child_form1 as normal form
Form child_form1 = form1.createChild("Forma Hija 1");

rad_main()

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es una forma MDI?	
¿Qué significa MDI?	
¿Cómo creamos una subforma dentro de la forma principal?	

Ejercicio 12 - Timers

```

/* ej12_Timers
Ejemplo del uso de timers */
#include <radc++.h>

Form form1("Prueba Timer - RAD C++ Ejemplo"); //create simple form

String timer1 = "Timer1"; //unique name of timer
int x=0;          //temporary variable to increase on timer notification

//create procedure for form to receive events
FormProcedure form1Proc(FormProcArgs) {

```

```

ON_CLOSE() { //close applciation when form is closed

    //remove timer when quitting application (remove all timers before exiting)
    form1.removeTimer(timer1);

    Application.close();
}
ON_TIMER_NOTIFICATION(timer1) { //when timer named 'Timer1' notifies
    //increase value of variable 'x' and assgn it to caption/title of form
    //NOTE : str returns string presentation of a numeric value
    form1.caption = str(++x);
}

return 0;
}

rad_main()

    //attach procedure to the form
    form1.procedure = form1Proc;

    //Set new timer, set notification time to 1 second, which will notify our Form after
    //each second continously until timer is not removed
    form1.setTimer(timer1, 1 * SECONDS);

    //MOREOVER:
    //1. you may pass direct value in terms of milliseconds e.g. 1000 in this case which is 1 second
    //2. you can use 2 * MINUTES which is 2 minutes, or 2 * HOURS which is repectively 2 hours

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué son los timers?	
¿Cómo especificamos un timer?	
¿Cómo activamos un timer?	
¿Cómo eliminamos un timer?	

Ejercicio 13 - Video

```

/* ej13_Video
Ejemplo de como usar un control de video para mostrarlo en la ventana */
#include "radc++.h"

Form form1("Video Control - RAD C++ Ejemplo");

//Create a container to play video in
// AUTO_ID is unique id of control (it is a macro)
// form is parent window of it
// 3rd argument is filename to play, see below

Video avi(AUTO_ID,form1,"clock.avi");

rad_main()

    avi.fitExact(); //fit exact to the parent form area

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
Describe la sintaxis del control de Video:	
¿Qué hace fitExact()?	

Ejercicio 14 - TableView

```

/* Ej14_TablaVistaGrid
Ejemplo para mostrar el control de tabla */
#include <radc++.h>

Form form1("TableView example - RAD C++ 1.2");
TableView t2("tbv",AUTO_ID,0,0,250,200,form1);

```

```

rad_main()

    t2.addColumn("Name");
    t2.addColumn("Email");
    t2.addColumn("Age");

    t2.addRow("Ali Imran");
    t2.addCell(1,0,"support@radcpp.com");
    t2.addCell(2,0,"29");

    t2.addRow("Otro Cliente");

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Para que sirve el control TableView?	
¿Cómo agregamos una columna al control?	
¿Cómo agregamos una fila al control?	

Ejercicio 15 - ScrollBars

```

/* ej15_ScrollBars
Ejemplo para el manejo de ScrollBars en una ventana */
#include <radc++.h>

Form    form1("ScrollBar 1 - RAD C++ Ejemplo");
ScrollBar scr(AUTO_ID,20, 20,20,200,form1,RCP_VERTICAL);

FormProcedure proc(FormProcArgs) {
    ON_SCROLL_CHANGE(scr) {
        LONG pos = scr.position; //get scrollbar position
        form1.caption = str(pos);
    }
}

```

```

        ON_CLOSE()
            Application.close();

        return 0;
    }

rad_main()

        form1.procedure = proc;

        //to set custom scrollbar range
        //scr.setRange(1,100); [min, max]

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un scrollbar?	
¿Cómo podemos hacer un scrollbar Horizontal?	
¿Cómo ponemos los limites para el scrollbar?	

Ejercicio 16 – ImageBox

```

/* ej16_ImagenExterna
Como mostrar una imagen dentro de una ventana */
#include <radc++.h>

Form form1("Mostrar imagen desde archivo - RAD C++ Ejemplo ");

ImageBox ibox(AUTO_ID,0,100,50,200,200,form1); //2nd arg : 0 = no image by default

rad_main()

        //Note : the image path must be absolute
        //if you wish to load from current folder, do not pass only image name,

```

```
// but use Application.path + "\\filename.bmp" instead.
//eg. C:\\ABC\\test.wmf

//ibox.loadImage(Application.path + "\\loadimage-ext.wmf"); //load from same folder
ibox.loadImage(Application.path + "\\pawprint.jpg"); //load from same folder
```

```
rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué hace el control ImageBox?	
¿Por qué necesitamos usar Application.path?	

Ejercicio 17 – ImageBox desde EXE

```
/* ej17_MostrarImagenExe
Mostrar una imagen desde un archivo EXE */
#include <radc++.h>

Form form1("Cargar imagen desde otro exe - RAD C++ Ejemplo");

ImageBox ibox(AUTO_ID,0,100,50,200,200,form1); //2nd arg : 0 = no image by default

rad_main()

    ibox.loadExternal("C:\\WINDOWS\\winhlp32.exe",1047);
    //1047 is ID of an image in the resources of winhlp32.exe

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es la diferencia entre este programa y el ejercicio 16?	

¿Por qué necesitamos usar esta forma de control?	
--	--

Ejercicio 18 – ImageBox desde Internet

```
/*ej18_MostrarImagenDeInternet
Como mostrar una imagen directamente desde internet */
#include <radc++.h>

Form form1("Mostrar imagen de internet - RAD C++ Ejemplo");

ImageBox ibox(AUTO_ID,0,50,50,300,200,form1); //2nd arg : 0 = no image by default

rad_main()

        ibox.loadImage("http://lamordida.net/wp-content/uploads/2011/03/Dev-Cpp-4.9.9.2.jpg");

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es la diferencia entre este programa y el ejercicio 16?	
¿Podemos usar este comando sin estar conectado a Internet?	

Ejercicio 19 – Información del Sistema

```
/* ej19_InfoSystem
Mostrar informacion del sistema */
#include <radc++.h>

Form form1("Informacion del Sistema y Aplicacion - RAD C++ Ejemplo");
RichTextBox txt("",AUTO_ID,0,0,10,10,form1);
```

```

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();
    ON_MOVE() {
        form1.text = str(Application.ticks/1000) + "," + str(System.ticks/1000);
    }

    return 0;
}

rad_main()

    form1.procedure = proc;

    txt.fitExact();
    txt.text =

        System.windowsFolder + "\n" +
        System.systemFolder + "\n" +
        System.currentFolder + "\n" +
        System.computerName + "\n" +
        System.userName + "\n" +
        System.windowsVersion + "\n" +
        Application.fileName + "\n" +
        Application.path + "\n" +
        Application.buildDate + "\n" +
        Application.buildTime + "\n"

    ;

rad_end()

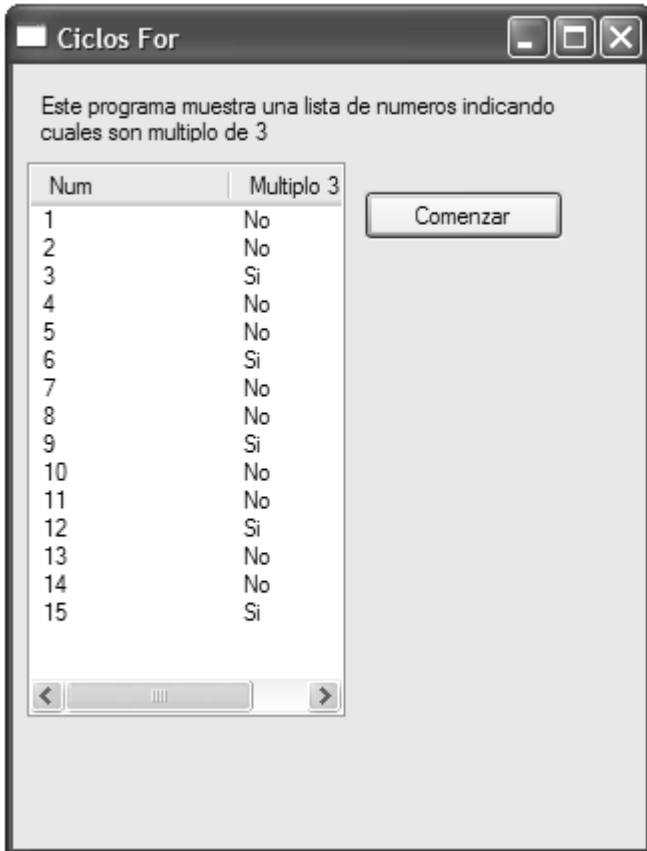
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué hace este programa?	
¿Qué controles usamos en este programa?	
¿Cómo diferenciamos las variables de sistema con las de la aplicación o programa?	

PRACTICA 6

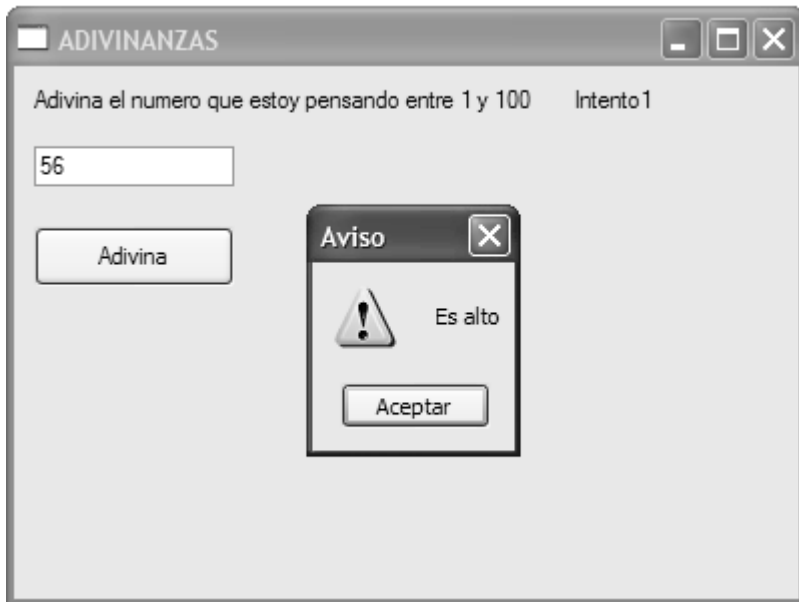
Realiza el ejercicio 19 sobre Ciclos for del Cuaderno de Ejercicios de Programacion I usando el control de TableView para hacerlo de manera grafica.



PRACTICA 7

Realiza el ejercicio 22a, que es el Juego de Adivinanzas del Cuaderno de Ejercicios de Programacion I, usando la librería grafica.

Usaras ademas las cajas de dialogo de msgBox, InfoBox, errorBox, warnBox. Preguntale a tu profesor.



PRACTICA 8

Realiza el ejercicio de Practica 1/Zodiaco del Cuaderno de Ejercicios de Programacion I usando la librería RADC++.

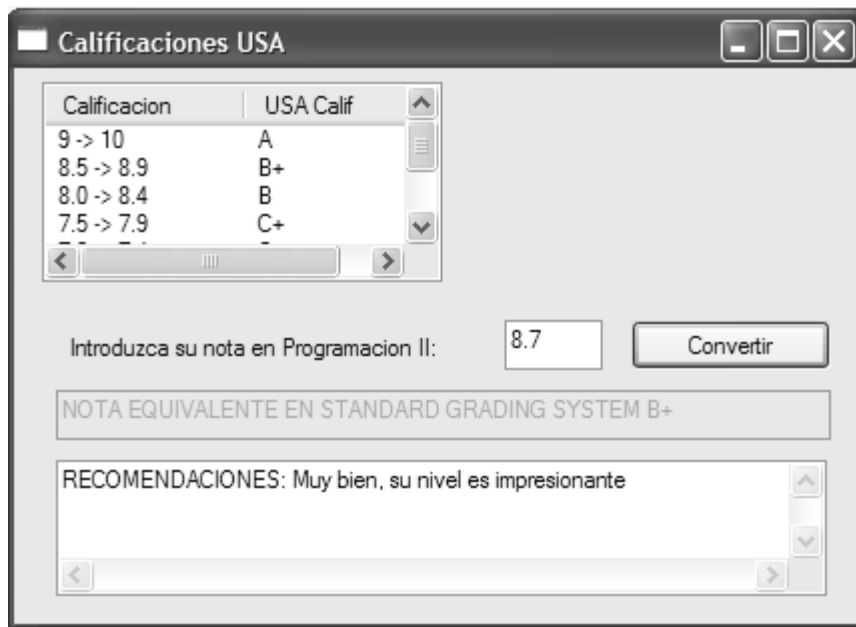
Como recordatorio, este ejercicio preguntaba al usuario su mes y dia de nacimiento e indicaba cual era el signo zodiacal al que pertenecia. Sin embargo, para esta practica, necesitaras tener las imágenes de los 12 simbolos del zodiaco, para que ademas de indicar el nombre, muestre la imagen correspondiente.



PRACTICA 9

Realiza el ejercicio de Practica 2/Calificaciones USA del Cuaderno de Ejercicios de Programacion I usando la librería RADC++

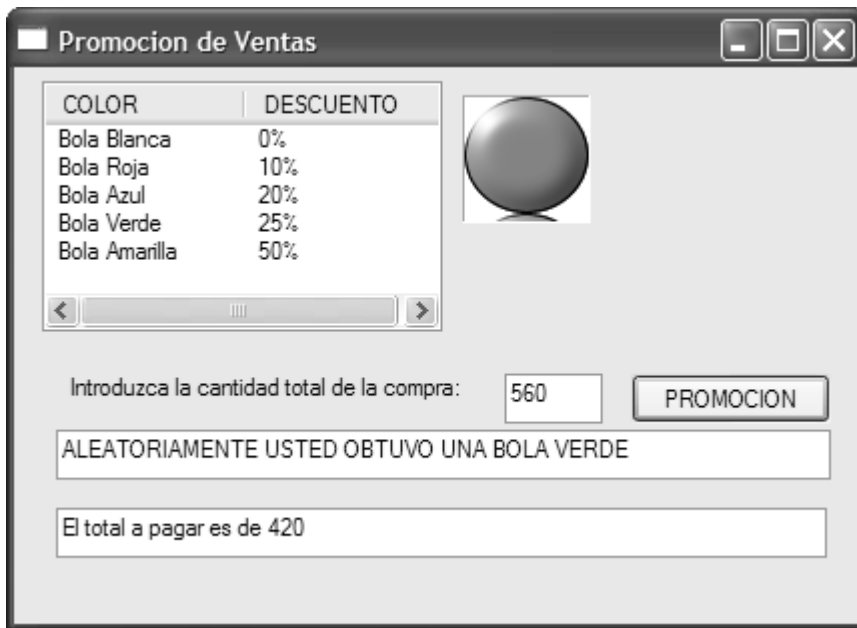
Como recordatorio, este ejercicio mostraba la tabla de equivalencias entre el sistema decimal usado para calificar con el sistema por letras de Estados Unidos. Preguntara al usuario su calificacion o nota en una materia, convertira dicha calificacion decimal al sistema de USA y lo indicara, asi como una recomendación o nota extra.



PRACTICA 10

Realiza el ejercicio de Practica 4/Ventas Descuentos del Cuaderno de Ejercicios de Programacion I usando la librería RADC++

Este ejercicio permite introducir el monto de una compra de un usuario, y luego de manera aleatoria selecciona una bola de un color que contendra un descuento. Asi la bola blanca, no tendra descuento, y la bola amarilla tendra un 50%. En base a este descuento, calcular el nuevo monto a pagar.



Ejercicio 20 - DragNDrop

/* ej20_DragDrop

Ejemplo del uso del evento para hacer Arrastre */

#include <radc++.h>

//create new form

Form form1("Arrastre algunos archivos y suelte - RAD C++ Ejemplo");

//create form1's control procedure

FormProcedure form1Proc(FormProcArgs) {

 ON_CLOSE() { //close applciation when form is closed

 Application.close();

 }

 ON_FILES_DROP() { //check if file(s) dropped on form

 //get dropped files list in variable droppedFiles

 StringList droppedFiles;

 getDroppedFiles(droppedFiles);

 //get total files dropped and assign to variable total

 Number total = droppedFiles.length;

 //combine file names with \r\n and assign to string 'files' to display on messagebox

```

        String files = droppedFiles.join("\r\n");

        //create new string 'title' for messagebox
        String title = "Total "+str(total)+" archivo(s) soltados en la forma";

        //display the message box now
        form1.msgBox(files,title);
    }

    return 0;
}

rad_main()

        //attach procedure 'form1Proc' with form1
        form1.procedure = form1Proc;

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué significa DragNDrop?	
¿Qué hace este programa?	
¿Cómo sabemos cuantos archivos se arrastraron?	

Ejercicio 21 – Tip del Dia

```

/* ej21_tipdeldia1
Como crear un tip del dia */
#include <radc++.h>

/* This program only writes the tips file, to use later */

Form form1("Escribiendo un archivo tip del dia - RAD C++ Ejemplo");

TipOfDay tod("", "", form1, false); //create invisible tip of day control

```

```

rad_main()

    //add 3 tips to control, allowe following HTML fields
    tod.addTip("I am tip 1 title", "I am tip 1 <B>description</B>");
    tod.addTip("I am tip 2 title", "I am tip 2 <I>description</I>");
    tod.addTip("I am tip 3 title", "I am tip 3 <U>description</U>");

    //now save them as new tip file to load later in application
    tod.saveFile("tipoftheday.rct");

    form1.infoBox("New tips file 'test.rct' has been created, click OK to exit");
    Application.close();

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es el objetivo del programa?	
¿Qué controles usamos en este programa?	
¿Dónde y como almacenamos la información del control?	

Ejercicio 22 - TipOfDay

```

/* ej22_TipDelDia2
Ahora mostraremos el tip del Dia antes de iniciar un programa
Necesitamos copiar el archivo .rct creado en el ejercicio anterior
a la carpeta donde creamos este programa para que funcione */
#include <radc++.h>

```

```
Form form1("Tip del Dia - RAD C++ Ejemplo");
```

```

//add tip of the day form to project
TipOfDay t("tipoftheday.rct", "Tip del Dia", form1);

```

```
//tipoftheday.rct is file that we create in last example
//"Tip of the Day" is title of tipofday form
//form1 is parent form

rad_main()

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es el objetivo del programa?	
¿Cuál es la diferencia con el ejercicio 21?	

Ejercicio 23 - Tab

```
/* ej23_TabControl
Para mostrar el uso del control Tab */
#include <radc++.h>

Form form1("Tab Control - RAD C++ Ejemplo");
//create new tab control
Tab          t("",AUTO_ID,0,0,form1.cwidth,form1.cheight,form1);
//add 3 pages to tab control
TabPage      p1=t.addPage("Pag1");
TabPage      p2=t.addPage("Pag2");
TabPage      p3=t.addPage("Pag3");

//4 controls to attach with tab pages

Form form2("Hijo de Tab",0,0,100,100,RCP_NONE,true,true,false,false,false,form1,true,0);

Button       b1 ("Boton para tab1", AUTO_ID,0,10,100,50,form2);
TableView    b12("1",          AUTO_ID,0,70,100,100,form2);
Button       b2("para tab 2",   AUTO_ID,0,0,10,10,form1);
RichTextBox  b3 ("Soy tab 3 anclado a objeto",  AUTO_ID,0,0,10,10,form1);
```

```
FormProcedure childformproc(FormProcArgs) {

    ON_RESIZE() {
        b1.fitToWidth();
        b12.fitToWidth();
    }
    ON_COMMAND_BY(b1) form1.msgBox("btn1");
    ON_COMMAND_BY(b12) form1.msgBox("btn2");

    return 0;
}

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    //now check of p1 and p2 of tabcontrol 't' are selected
    //Even if not processing events, just put semi collin after,
    //but make sure to write them down to show attached Objects properly
    ON_TAB_SELECT(t,p1); //t is tab control p1 is tab page
    ON_TAB_SELECT(t,p2);
    ON_TAB_SELECT(t,p3);

    ON_RESIZE() { //when form is resized
        //resize tab control and fit to entire form area
        t.fitExact();

        //also resize and adjust attached objects of pages, each individually
        p1.adjust();
        p2.adjust();
        p3.adjust();
    }

    return 0;
}

rad_main()

form1.procedure = proc;
form2.procedure = childformproc;

b12.addColumn("Nombre");
```

```

b12.addColumn("Email");
b12.addColumn("Edad");

b12.addRow("abcd");
b12.addRow("abcd");
b12.addRow("abcd");
b12.addRow("abcd");

p1.attachObject(form2);
p2.attachObject(b2);
p3.attachObject(b3);

//since p1 is first tab, so it's contents need to be adjusted manually (just once)
p1.adjust();

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es lo que hace el programa?	
¿Qué controles usamos aquí?	
¿Por qué es necesario usar este control de pestañas?	
¿Cómo sabemos en que pestaña estamos?	

Ejercicio 24 – Escribir archivos Excel

```

/* ej24_EscribirXLS
Ejemplo para escribir archivos de Excel */
#include <radc++.h>

Form form1("Microsoft Excel Escritura de archivo - RAD C++ Ejemplo");

rad_main()

XLS x; //create variable of class XLS

```

```
//Row and column must be ZERO based
```

```
x.addCell(0,1,"name:");          x.addCell(1,1,"Ali Imran");
x.addCell(0,2,"email:");        x.addCell(1,2,"support@radcpp.com");
x.addCell(0,3,"age:");          x.addCell(1,3,"30");
```

```
x.addCell(0,10,"RAD C++ XLS class test");
```

```
x.writeFile("myfile.xls");
```

```
form1.msgBox("File has been written");
Application.close();
```

```
rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es lo que hace el programa?	
¿Qué controles usamos aquí?	
¿Cómo grabo la informacion?	

Ejercicio 25 – System Tray

```
/* ej25_SystemTray
```

```
Ejemplo para demostrar como se puede usar el systemTray en las
aplicaciones */
```

```
#include <radc++.h>
```

```
Form form1("Poner y cambiar el icono en System Tray - RAD C++ Ejemplo");
```

```
Label label("Por favor hage click en el nuevo icono para agreagrlo a su bara de tareas System Tray (creado en
este ejemplo), el cual cambiara el icono y el tooltip."
```

```
,-1,0,0,300,50,form1);
```

```
//Define 2 new objects of type Icon to hold our icons
```

```
//NOTE: IDI_APPLICATION and IDI_EXCLAMATION are predefined by system
```

```
Icon icon1(IDI_APPLICATION);
```

```
Icon icon2(IDI_EXCLAMATION);
```

```
//create procedure for form to receive events
FormProcedure form1Proc(FormProcArgs) {

    ON_CLOSE() { //close application when form is closed

        form1.removeTrayIcon(); //remove the icon we set when quitting application
        Application.close();

    }

    ON_TRAYICON_CLICK(form1) { //check if the tray icon has been clicked

        //change the icon and the tooltip of icon that we set in system tray
        //first argument is the icon (icon2) and second is tooltip of icon in tray
        form1.resetTrayIcon(icon2,"Tray icon tooltip cambiado");

        //display a message box
        form1.msgBox("Note en el system tray! el icono que ha clickeado ha sido cambiado igualmente
que el tooltip...");

    }

    return 0;
}

rad_main()

    //attach procedure to the form
    form1.procedure = form1Proc;

    //set the icon1 as system tray icon for this form
    //icon1 is the icon to be set and 2nd argument is the default tooltip
    form1.setTrayIcon(icon1,"Tray icon tooltip");

    label.center(); //center label in form

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es el objetivo del programa?	
¿Cómo definimos un objeto Icono?	
¿Cuál es el evento asociado al system tray en la forma?	
¿Cómo cambio de icono y tooltip?	
¿Qué es un tooltip?	

Ejercicio 26 – Subclase Boton

Responde a lo siguiente:

Pregunta	Respuesta

Ejercicio 27 – Subclase ListBox

```
/* ej27_Subclase ListBox
```

```
Ejemplo de subclase para el control ListBox */
```

```
#include <radc++.h>
```

```
Form form1("Subclass listbox - RAD C++ Example");
```

```
ListBox l("", AUTO_ID,0,0,100, 100,form1);
```

```
WNDPROC ActualListBoxProcedure=NULL;
```

```
FormProcedure form1Proc(FormProcArgs) {
```

```
    ON_CLOSE() Application.close();
```

```
    return 0;
```

```
}
```

```
//subclass procedure
```

```
ControlProcedure CustomListBoxProcedure(HWND hwnd, UINT message, WPARAM wParam, LPARAM lParam)
```

```

{
    ON_LEFT_CLICK()
        form1.text = "ON_LEFT_CLICK";

    ON_LEFT_CLICK_RELEASE()
        form1.text = "ON_LEFT_CLICK_RELEASE";

    ON_MOUSEMOVE()
        form1.text = "ON_MOUSEMOVE";

    return CallWindowProc(ActualListBoxProcedure, hwnd, message, wParam, lParam);
}

rad_main()

    form1.procedure = form1Proc;
    ActualListBoxProcedure = (WNDPROC)l.setWProcedure(CustomListBoxProcedure);

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta

Ejercicio 28 – Splitter Vertical

```

/* ej28_SplitterVertical
Ejemplo de control Splitter en formato Vertical */
#include <radc++.h>

Form form1("Splitter Vertical - RAD C++ Ejemplo");

Splitter s(AUTO_ID,form1);

TextBox t1("Una caja de texto",AUTO_ID,0, 0,100,10,form1);
Button t2("Un boton", AUTO_ID,105,0,200,10,form1);

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_SPLITTER_CHANGE(s) { //s is splitter variable

```

```

        t1.width = splitPosition();
        t2.left = splitPosition()+5;
        t2.width = form1.cwidth - (t1.width+5);
    }
    ON_RESIZE( ){
        t1.fitToHeight();
        t2.fitToHeight();
        t2.width = form1.cwidth - (t1.width+5);
        s.fitToHeight();
        s.setBounds(50,form1.cwidth-50);
    }

    return 0;
}

rad_main()

    s.setPosition(100);
    form1.procedure = proc;
    s.refresh();

    form1.resize(500,400); //resize form
    form1.center();      //center form on screen

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un splitter?	
¿Qué eventos usamos aquí y que hacen?	
¿Cómo redimensionamos una forma?	

Ejercicio 29 – Splitter Horizontal

```
/* ej29_SplitterHorizontal
Otro ejemplo de splitter, pero esta vez de manera horizontal */
#include <radc++.h>

Form form1("Horizontal Splitter - RAD C++ Ejemplo");
Splitter s(AUTO_ID,form1,RCP_HORIZONTAL);
TextBox t1("Una caja de texto",AUTO_ID,0, 0,10,100,form1);
Button t2("Un boton", AUTO_ID,0,105,10,10 ,form1);

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_SPLITTER_CHANGE(s) {
        t1.height = splitPosition(); //macro that gets current moving splitter's position
        t2.top = splitPosition()+5;
        t2.height = form1.cheight - (t1.height+5);
    }
    ON_RESIZE( ) { //adjust all objects wen form is resized
        t1.fitToWidth();
        t2.fitToWidth();
        t2.height = form1.cheight - (t1.height+5);
        s.fitToWidth();
        s.setBounds(50,form1.cheight-50);
    }

    return 0;
}

rad_main()

    s.setPosition(100); //set splitter's position
    form1.procedure = proc;
    s.refresh();

    form1.resize(500,250);
    form1.center();

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es la diferencia con el ejercicio 28?	

Ejercicio 30 - Menus

```
/* ej30_Menus
Ejemplo para incluir menus en nuestra forma */
#include <radc++.h>

Form form1("Menu - RAD C++ Ejemplo");

Menu menu(form1);
MenuItem __File,__Exit,__Help,__About;

FormProcedure form1Proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_COMMAND_BY(__Exit) Application.close();
    ON_COMMAND_BY(__About)
    form1.infoBox("Ejemplo de Menu - RAD C++ 1.2\r\n\r\nby www.radcpp.com");

    return 0;
}

rad_main()

    form1.procedure = form1Proc;
//main menus
    __File = menu.add("&Archivo");
    __Help = menu.add("A&yuda");

    //sub-menus not tracked
    __File.add("&Nuevo");
    __File.add("&Abrir");
    __File.add("&Guardar");
    __File.addSeparator();
```

```
//sub-menus tracked
    __Exit = __File.add("&Salida",AUTO_ID);
    __About = __Help.add("&Acerca de",AUTO_ID);

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué hace el control Menu?	
¿Cómo estableces las opciones principales?	
¿Cómo se crean las subopciones de un menú?	
¿Para que sirve el & en la creación de las opciones?	

PRACTICA 11

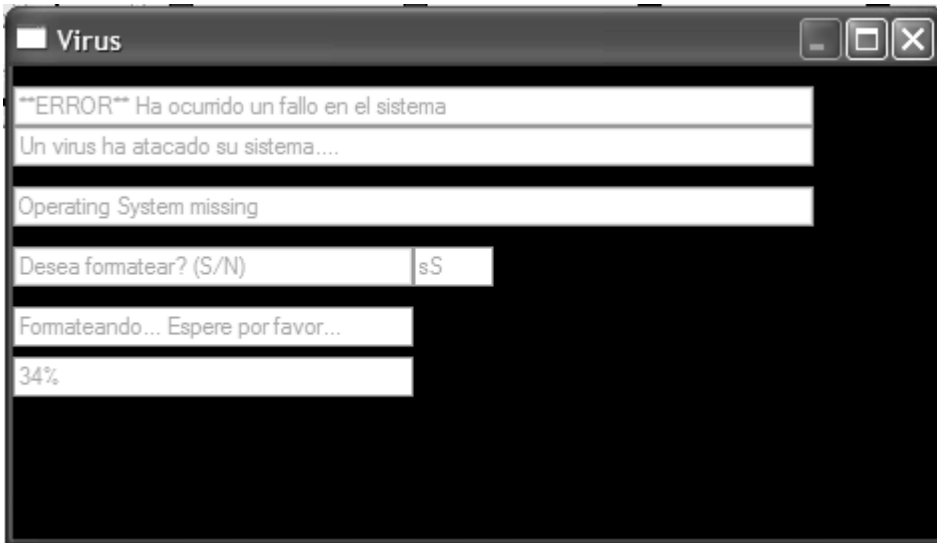
Realiza la simulacion de un virus usando los comandos para la consola.

Basicamente tiene que crear una ventana al tamaño mas grande, de color negro (o azul de fondo), poniendo algun texto que intimide al usuario, por ejemplo:

```
** ERROR ** Ha ocurrido un fallo en el sistema
Un virus ha atacado su sistema
Operating System missing
Desea formatear? S
```

Y cuando el usuario responda que No o cualquier otra tecla, de todas formas proceda a una simulacion de formateo del disco duro. Para esto, necesitara usar un TIMER de la forma, para que en la pantalla aparezca:

Formateando 1%, 5%, etc. Y que cuando termine el formateo, esto es llegando a 100, la maquina se reinicie de manera automatica asustando asi al usuario aun mas. Por supuesto, puede simplemente apagarla, salir o cambiar de sesion usando alguno de los comandos de Sistema.



PRACTICA 12

Es hora de hacer otro programa de juego para una vez que se concluya nos divertamos un rato. Realice el programa del Juego del Gato (Tres en raya o TicTacToe) según el cuaderno de ejercicios de Programacion I, en el ejercicio 29.

Aunque debiera cambiar un poco la logica del programa ya que la interfaz grafica es un poco mas sencilla. Necesitara hacer 9 cajas de texto que es donde el usuario pondra los valores de X y O. Siempre en mayusculas. El programa debiera detectar cada que el usuario pone una letra si ya gano el juego o no.

Tambien es conveniente que use un boton para poner las instrucciones indicandole al usuario que solo debe usar letras mayusculas.



Ejercicio 31 - MiniNotepad

```

/* ej31_MiniNotepad
Un ejemplo para crear un pequeño bloc de notas */
#include <radc++.h>

/*
    It is suggested that you try understanding Simple Form and Menus examples first
    and then get to this example.

    NOTE: This program may crash if file size is larger than 65535 (65.5 kb), because
    richtextbox by default accomodates 65535 bytes. Use RichTextBox::setLimit() to accomodate
    number of characters.
*/

Form form1("Ejemplo Mini NotePad - RAD C++ Ejemplo");

Menu menu(form1);
MenuItem __File,__Open,__Save,__Exit,__Help,__About;
RichTextBox txt("", AUTO_ID, 0, 0, 0, 0, form1);

FormProcedure form1Proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_COMMAND_BY(__Exit) Application.close();
    ON_COMMAND_BY(__About)
form1.infoBox("Ejemplo Mini NotePad - RAD C++ 1.2\r\n\r\nby www.radcpp.com");

    ON_COMMAND_BY (__Open) {
        if(form1.open()) {           //yes user selected file
            BinaryData bin;
            bin.loadFile(form1.fileName); //load selected file
            txt.text = bin.c_str();      //get NULL terminated string only
            bin.clear();                //release the memory
            form1.caption = form1.fileName;
        }
    }
    ON_COMMAND_BY (__Save) {
        if(form1.save()) {           //yes user selected file to save
            BinaryData bin;
            String data = txt.text;
            bin.add((UCHAR *)data.c_str(),data.length);    //get NULL terminated string and save
to file

```

```

        bin.writeFile(form1.fileName);
        bin.clear();           //release the memory
        form1.caption = form1.fileName;
    }
}
ON_RESIZE() txt.fitExact();
return 0;
}

rad_main()

        form1.procedure = form1Proc;
        txt.fitExact();

//main menus
        __File = menu.add("&Archivo");
        __Help = menu.add("A&yuda");

//sub-menus tracked
        __Open = __File.add("&Abrir",AUTO_ID);
        __Save = __File.add("&Guardar",AUTO_ID);
        __File.addSeparator();
        __Exit = __File.add("&Salida",AUTO_ID);
        __About = __Help.add("&Acerca de...",AUTO_ID);

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es el objetivo del programa?	
¿Qué controles usamos?	
¿Cómo grabamos un archivo?	
¿Cómo abrimos un archivo?	

Ejercicio 32 - Formas

```

/* ej32_Formas
Para manejar el color de fondo de las formas y controles */
#include <radc++.h>
Form form1("Color de fondo Forma/control - RAD C++ Ejemplo");

TextBox txt("Soy una caja de texto",AUTO_ID,100,100,100,100, form1);

FormProcedure proc (FormProcArgs) {
    ACCEPT_COLORS();

    ON_CLOSE() Application.close();

    return 0;
}

rad_main()
    form1.procedure = proc;
    form1.backgroundColor = 0xffffffff; //pure white
    txt.backgroundColor = 0xff0000;    //blue
rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es el objetivo del programa?	
¿Qué controles usamos?	
¿Cómo grabamos un archivo?	
¿Cómo abrimos un archivo?	

Ejercicio 33 - ListBox

```
/* ej33_ListBox
Ejemplo del control ListBox */
#include <radc++.h>

Form Form1("List Box ejemplo - RAD C++ Ejemplo");
ListBox list("", AUTO_ID, 10, 15, 100, 200, Form1); //create new list box

Button btn_add("< Agregar esto <", AUTO_ID, 120, 15, 100, 25, Form1);
TextBox txt_newval("Nuevo Valor", AUTO_ID, 230, 15, 150, 25, Form1);

Label label("Selecciona Entrada", AUTO_ID, 120, 45, 260, 15, Form1);
ReadOnlyBox txt_selected("", AUTO_ID, 120, 60, 260, 25, Form1);

Button btn_remove("[ X ] Remover seleccion", AUTO_ID, 120, 130, 260, 25, Form1);
Button btn_select("Progrmaticamente seleccionar item 2", AUTO_ID, 120, 160, 260, 25, Form1);

FormProcedure procedure1 (FormProcArgs); //prototype of form procedure

rad_main()

    //attach form procedure
    Form1.procedure = procedure1;

    //add some enteries to listbox
    list.add("entrada 1");
    list.add("entrada 2");
    list.add("entrada 3");
    list.select(0); //select first entry

rad_end()

FormProcedure procedure1 (FormProcArgs) { //implementation of form procedure

    ON_CLOSE() {
        //exit application
        Application.close();
    }
    //when an item is selected, put its text in textbox txt_selected
    ON_ITEMSELECT(list) {
        txt_selected.text = list[list.selectedItemIndex];
    }
}
```

```

    ON_COMMAND_BY(btn_add) {
        //add new value of textbox txt_newval ot listBox
        list.add(txt_newval.text);
    }
    ON_COMMAND_BY(btn_remove) {
        //remove selected item in listBox
        list.remove(list.selectedIndex);
    }
    ON_COMMAND_BY(btn_select) {
        //remove selected item in listBox
        list.select(1); //1 means item 2, items have ZERO based indeses
    }
    return 0;
}

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un listBox?	
¿Cómo agregamos elementos al control?	
¿Cómo sabemos que elemento de la lista esta seleccionado?	

Ejercicio 34 - ComboBox

```

/* ej34_ComboBox
Ejemplo de control ComboBox */
#include <radc++.h>

Form Form1("Combo Box ejemplo - RAD C++ Ejemplo");
ComboBox combo("", AUTO_ID, 10, 15, 100, 200, Form1); //create new combo box

Button btn_add("< Agregar esto <", AUTO_ID, 120, 15, 100, 25, Form1);
TextBox txt_newval("Nuevo valor", AUTO_ID, 230, 15, 150, 25, Form1);

Label label("Entrada Seleccionada", AUTO_ID, 120, 45, 260, 15, Form1);
ReadOnlyBox txt_selected("", AUTO_ID, 120, 60, 260, 25, Form1);

Button btn_remove("[ X ] Remover seleccion", AUTO_ID, 120, 130, 260, 25, Form1);
Button btn_select("Progrmaticamente seleccionar item 2", AUTO_ID, 120, 160, 260, 25, Form1);

```

```
FormProcedure procedure1 (FormProcArgs); //prototype of form procedure

rad_main()

    //attach form procedure
    Form1.procedure = procedure1;

    //add some enteries to combobox
    combo.add("entrada 1");
    combo.add("entrada 2");
    combo.add("entrada 3");
    combo.select(0); //select first entry

rad_end()

FormProcedure procedure1 (FormProcArgs) { //implementation of form procedure

    ON_CLOSE() {
        //exit application
        Application.close();
    }
    //when an item is selected, put its text in textbox txt_selected
    ON_ITEMSELECT(combo) {
        txt_selected.text = combo[combo.selectedItemIndex];
    }
    ON_COMMAND_BY(btn_add) {
        //add new value of textbox txt_newval ot combobox
        combo.add(txt_newval.text);
    }
    ON_COMMAND_BY(btn_remove) {
        //remove selected item in combobox
        combo.remove(combo.selectedItemIndex);
    }
    ON_COMMAND_BY(btn_select) {
        //remove selected item in combobox
        combo.select(1); //1 means item 2, items have ZERO based indeses
    }
    return 0;
}
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un ComboBox?	

Ejercicio 35 – ListBox MultiColumna

```

/* ej35_MultiColumnListBox
Ejemplo para mostrar el control ListBox con mas de una columna */
#include <radc++.h>

Form Form1("Multi-Columna List Box ejemplo - RAD C++ Ejemplo");
MCListBox list("", AUTO_ID, 10, 15, 370, 100, Form1); //create new mc list box

Button btn_add("^ Agregar esto <", AUTO_ID, 120, 115, 100, 25, Form1);
TextBox txt_newval("nuevo", AUTO_ID, 230, 115, 150, 25, Form1);

Label label("Entrada Seleccionada", -1, 120, 145, 260, 15, Form1);
ReadOnlyBox txt_selected("", AUTO_ID, 120, 160, 260, 25, Form1);

Button btn_remove("[ X ] Remover item seleccionado", AUTO_ID, 120, 200, 260, 25, Form1);
Button btn_select("Progmaticamente seleccionar item 2", AUTO_ID, 120, 230, 260, 25, Form1);

FormProcedure procedure1 (FormProcArgs); //prototype of form procedure

rad_main()

    //attach form procedure
    Form1.procedure = procedure1;

    //add some enteries to listbox
    for(int i=1; i<=40; i++)
        list.add("item "+str(i));

    list.select(0); //select first entry
    list.setColumnWidth(50);

rad_end()

FormProcedure procedure1 (FormProcArgs) { //implementation of form procedure

```

```

    ON_CLOSE() {
        //exit application
        Application.close();
    }
    //when an item is selected, put its text in textbox txt_selected
    ON_ITEMSELECT(list) {
        txt_selected.text = list[list.selectedItemIndex];
    }
    ON_COMMAND_BY(btn_add) {
        //add new value of textbox txt_newval ot listbox
        list.add(txt_newval.text);
    }
    ON_COMMAND_BY(btn_remove) {
        //remove selected item in listbox
        int selection=list.selectedItemIndex;
        if(selection== -1)
        Form1.warnBox("No item seleccionado!");
        list.remove(selection);
    }
    ON_COMMAND_BY(btn_select) {
        //remove selected item in listbox
        list.select(1); //1 means item 2, items have ZERO based indeses
    }
    return 0;
}

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un ListBox MultiColumna?	

Ejercicio 36 – ComboBox con Iconos

/*ej36_ComboBoxIcons

Ejemplo para crear un control ComboBox con Iconos */

#include <radc++.h>

Form Form1("Combo Box con iconos ejemplo - RAD C++ Ejemplo");

ComboBox combo("", AUTO_ID, 10, 15, 200, 100, Form1); //create new combo box

```
FormProcedure procedure1 (FormProcArgs) {
    ON_CLOSE() Application.close();
    return 0;
}
IconList icl; //create new iconlist
Icon icon1(IDI_APPLICATION), //create 3 icon objects
    icon2(IDI_EXCLAMATION),
    icon3(IDI_HAND);

rad_main()

    //attach form procedure
    Form1.procedure = procedure1;

    //add icons to icon list
    icl.add(icon1);
    icl.add(icon2);
    icl.add(icon3);

    //attach iconlist to combobox
    combo.setIconList(icl);

    //add some enteries to combobox
    //note: 2nd argument is reserved and msut be -1
    combo.add("entrada 1",-1,0); //3rd argument is ZERO basded index of iconin list
    combo.add("entrada 2",-1,1);
    combo.add("entrada 3",-1,2);
    combo.add("entrada con icono por default"); //did nto pass last 2 arguments so first icon is
selected
    combo.select(0); //select first entry

    combo.center(); //center the combo onto form

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es el objetivo del programa?	
¿Qué hace el tipo de dato IconList?	
¿Cómo agreamos los iconos al control combobox?	

Ejercicio 37 - ListView

```

/* ej37_ListView
Ejemplo de control ListView */
#include <radc++.h>

Form Form1("List View ejemplo - RAD C++ Ejemplo");
Label      label("Item de texto Seleccionado:", -1, 50, 10, 300, 25, Form1);
ReadOnlyBox text1("", -1, 50, 40, 300, 25, Form1);
ListView   listv("", AUTO_ID, 50, 75, 300, 160, Form1); //create new listview

FormProcedure procedure1 (FormProcArgs) {
    ON_CLOSE() Application.close();
    ON_OBJECTSELECT(listv) { //item has been selected in listview
        int selection = listv.selectedItemIndex;
        if(selection == -1) //nothing selected, it was just some other event
            text1.text = "Actualmente sin seleccion!";
        else
            text1.text = listv[listv.selectedItemIndex];
    }
    return 0;
}

//create new iconlist, RCP_LARGE becuase listview accepts large icons
IconList icl(RCP_LARGE);
//create some icon objects
Icon icon1(IDI_HAND), icon2(IDI_ASTERISK), icon3(IDI_APPLICATION), icon4(IDI_EXCLAMATION) ;

rad_main()

    //attach form procedure
    Form1.procedure = procedure1;

    //add icons to icon list

```

```

icl.add(icon1);
icl.add(icon2);
icl.add(icon3);
icl.add(icon4);

//now attach iconlist to listview
listv.setIconList(icl);

    //add some enteries to listview
    //note: 2nd argument is reserved and msut be empty string ""
listv.add("entrada 1","",1); //3rd argument is ZERO basded index of icon in list
listv.add("entrada 2","",2);
listv.add("entrada 3","",3);
listv.add("a. con icono default"); //did not pass last 2 arguments so first icon is selected
listv.add("b. con icono default"); //did not pass last 2 arguments so first icon is selected

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Cuál es la diferencia entre el ListView y el ListBox?	
¿Cuál es la diferencia entre el TableView y el ListView?	
¿Cómo asociamos los iconos con el control?	
¿Cómo agregamos los iconos al control?	
¿Cómo sabemos que item o icono se ha seleccionado?	

Ejercicio 38 – Icono de Forma

```

/* ej38_FormIcon
Ejemplo para cambiar el icono asociado a la forma */
#include <radc++.h>

Form Form1("Cambiar/Poner el Icono de la Forma - RAD C++ Ejemplo");
Button button1("Click a mi para cambiar el icono de la forma", AUTO_ID, 50, 10, 300, 35, Form1);

```

```
//create some icon objects
//NOTE: IDI_HAND and IDI_EXCLAMATION are predefined by system
Icon icon1(IDI_EXCLAMATION);
Icon icon2(IDI_HAND);

FormProcedure procedure1 (FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_COMMAND_BY(button1) {
        //set new icon to form's title bar
        Form1.icon = icon2;
    }

    return 0;
}

rad_main()

    Form1.procedure = procedure1;

    //set default icon to form's title bar
    Form1.icon = icon1;

    //center the button on form
    button1.center();

rad_end()
```

Ejercicio 39 – Slider Horizontal

```
/* ej39_HorizontalSlider
Ejemplo para mostrar el control Slider de manera horizontal */
#include <radc++.h>

Form form1("Slider Horizontal - RAD C++ Ejemplo");

Label label1("Por favor arrastre y mueva el slider con el raton",-1,100,20 ,200,30,form1);
Track track(AUTO_ID,100,60,200,30,form1); //create trackbar / slider
```

```
Label label2("Introduzca el numero de ticks",-1,100,110 ,200,20,form1);
NumberBox txt_num("15",AUTO_ID,100,135 ,200,20,form1);

Label label3("Introduzca una nueva posicion",-1,100,165 ,200,20,form1);
NumberBox txt_pos("0",AUTO_ID,100,190 ,200,20,form1);
```

```
FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    //user moved trackbar
    ON_TRACK_CHANGE(track) {
        int pos = track.position; //get track position
        label1.caption = str(pos);
    }

    ON_TEXT_CHANGED(txt_num) {
        //set new number of ticks
        track.ticks = val(txt_num.text);
    }

    ON_TEXT_CHANGED(txt_pos) {
        //set new number position
        track.position = val(txt_pos.text);
    }

    return 0;
}
```

```
rad_main()

    form1.procedure = proc;

    //set minmum and maximum range
    track.minRange = -15;
    track.maxRange = 15;
```

```
rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un slider?	
¿Cómo establecemos los valores minimos y maximos del control?	

Ejercicio 40 – Slider Vertical

```
/* ej40_VerticalSlider
Ejemplo del mismo control Track pero ahora vertical */
#include <radc++.h>

Form form1("Slider Vertical - RAD C++ Ejemplo");

Label label1("Por favor arrastre y mueva el slider con el raton",-1,100,20,200,30,form1);
//create trackbar / slider
Track track(AUTO_ID,100,60,30,150,form1,0,true,true,false,true);
//check documentation for track constructor arguments,
//8th argument is default number of ticks
//12th argument is bool _vertical, we passed true to make it vertical

Label label2("Introduzca nuevo numero de ticks",-1,150,60,150,20,form1);
NumberBox txt_num("15",AUTO_ID,150,85,150,20,form1);

Label label3("Introduzca nueva posicion",-1,150,155,150,20,form1);
NumberBox txt_pos("0",AUTO_ID,150,180,150,20,form1);

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    //user moved trackbar
    ON_TRACK_CHANGE(track) {
        int pos = track.position; //get track position
        label1.caption = str(pos);
    }

    ON_TEXT_CHANGED(txt_num) {
        //set new number of ticks
        track.ticks = val(txt_num.text);
    }
}
```

```
    ON_TEXT_CHANGED(txt_pos) {
        //set new number position
        track.position = val(txt_pos.text);
    }

    return 0;
}

rad_main()

    form1.procedure = proc;

    //set minmum and maximum range
    track.minRange = -15;
    track.maxRange = 15;

rad_end()
```

Ejercicio 41 - CheckBox

```
/* ej41_CheckBox
Ejemplo del control CheckBox */
#include <radc++.h>

Form    form1("Casilla verificacion - RAD C++ Ejemplo");
CheckBox check("Check box - click", AUTO_ID,100,100,200,20,form1);

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_CHECK(check) {
        form1.caption = "Marcado";
    }

    ON_UNCHECK(check) {
        form1.caption = "Desmarcado";
    }

    //NOTE : if you need to know about any message use ON_CHECKBOX_EVENT(check)

    return 0;
}
```

```

}

rad_main()

    form1.procedure = proc;

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un CheckBox o para que sirve?	
¿Cómo sabemos cuando la opcion esta marcada?	
¿Cómo sabemos cuando la opcion es falsa?	

Ejercicio 42 - RadioButtons

```

/* ej42_RadioButton
Ejemplo del control RadioButton */
#include <radc++.h>

Form    form1("Verificaciones - RAD C++ Ejemplo");

RadioButton radio1("Boton Radio 1 - seleccioname", AUTO_ID,100,100,200,20,form1);
RadioButton radio2("Boton Radio 2 - seleccioname", AUTO_ID,100,130,200,20,form1);

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_RADIO_EVENT(radio1)
        form1.caption = radio1.selected ? radio1.caption : radio2.caption;

    ON_RADIO_EVENT(radio2)
        form1.caption = radio2.selected ? radio2.caption : radio1.caption;

    return 0;
}

```

```
rad_main()

    form1.procedure = proc;

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un RadioButton o para que sirve?	
¿Cómo sabemos cuando la opcion esta marcada?	

Ejercicio 43 - ProgressBar

```
/* ej43_ProgressBar
Ejemplo para usar el control ProgressBar */
#include <radc++.h>

Form    form1("Progres Bar - RAD C++ Ejemplo");

ProgressBar pgb1(-1,100,50,200, 20,form1,RCP_HORIZONTAL); //horizontal
ProgressBar pgb2(-1,100,80, 20,100,form1,RCP_VERTICAL); //vertical

Label    lbl("Introduzca porcentaje",AUTO_ID,180,100,120,20,form1);
NumberBox txt("50",AUTO_ID,180,120,120,25,form1);
Button    btn("Ponga Porcentaje %",AUTO_ID,180,155,120,25,form1);

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_COMMAND_BY(btn) { //set percentage as per value of txt.text
        pgb1.percent=val(txt.text);
        pgb2.percent=val(txt.text);
    }

    return 0;
}
```

```
rad_main()

    form1.procedure = proc;

    //set numberbox digits limit
    txt.setLimit(3); //3 digits

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un ProgressBar o para que sirve?	
¿Cómo indicamos el porcentaje de avance?	

Ejemplo 44 - StatusBar

```
/* ej44_StatusBar
Ejemplo del control StatusBar */
#include <radc++.h>

Form    form1("StatusBar - RAD C++ Ejemplo");
StatusBar status("Parte 1 texto;Parte 2 texto",AUTO_ID,form1,2); //2 parts

Label   lbl1("Introduzca texto para las partes, delimite con ';'",-1,100,35,200,20,form1);
TextBox txt1("Part 1 text;Part 2 text",AUTO_ID,100,60,200,20,form1);

Label   lbl2("Introduzca nuevo numero de partes",-1,100,110,200,20,form1);
NumberBox txt2("2",AUTO_ID,100,135,50,20,form1);

Label   lbl3("Introduzca parte # para mostrar",-1,100,165,200,20,form1);
NumberBox txt3("0",AUTO_ID,100,185,50,20,form1);

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_RESIZE() { //when form is resized, adjust the statusbar
        status.adjust();
    }
}
```

```
ON_TEXT_CHANGED(txt1) { //change text of parts
    status.setText(txt1.text);
}

ON_TEXT_CHANGED(txt2) {
    int num = val(txt2.text);
    if(num > 0) {
        status.setParts(num); //set new number of parts
        //NOTE: application can crash if number of new parts is supplied ZERO

        status.setText(txt1.text); //remake popout effects
    }
}
ON_TEXT_CHANGED(txt3) {
    status.popOut(val(txt3.text),true);
}

return 0;
}

rad_main()

    form1.procedure = proc;

    txt2.setLimit(2);

    //1.You can use single string having values delimited with ';' semicolon
    //status.setText("Part 1 text;Part 2 text");

    //2. Second way to set text in statusbar parts
    //StringList list;
    //list.add("Part 1 text");
    //list.add("Part 2 text");
    //status.setText(list);

    //3.OR you can set text of certain part individually as:
    //status.setText(0,"some text");

    String part1_text = status[0]; //access the part 1 text, ZERO based index

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un StatusBar o para que sirve?	
¿Cómo indicamos las divisiones que queremos?	

PRACTICA 13

Desarrolla un programa que esta basado en la practica 7 del Cuaderno de Ejercicios de Programacion I. Este programa ira preguntando por la cantidad de billetes y monedas de cada valor que tiene el usuario y luego determinara la suma de dinero que hay en monedas y la suma que hay en billetes. El programa dira la suma total de dinero que tiene. Finalmente el programa dara al usuario la posibilidad de transformar la cantidad de la moneda local a dolares o euros según decida el usuario.

Deberas usar los controles de ListBox, Botones, Cajas de Texto, asi como de Tab (pestaña) para que en una pestaña sumes los valores y en la otra puedas hacer la conversion.

El tipo de cambio lo puedes especificar a tu gusto. Ej. Dólar: Compra 11.78 Venta 12.00, Euro Compra 16.05 Venta 16.90

Contando Dinero

Calcular Convertir

Billetes		Monedas	
Veinte		Uno	
Cincuenta		Dos	
Cien		Cinco	
Doscientos		Diez	
Quinientos			
Mil			
Total		Total	
TOTAL			0

Calcular

Contando Dinero

Calcular Convertir

Monto

PRACTICA 14

Realiza un programa de Agenda Telefonica con solamente el modulo de Altas. Es decir, que permita ingresar los datos a un archivo de texto y que los puedas visualizar en la pantalla.

He aquí algunos tips que puedes usar para realizar esta practica.

- 1) Usar MDI FORM
- 2) Crear una subventana para el modulo de altas
- 3) Crear un menu en la parte superior
- 4) Campos que podrias usar: Nombre, Apellido Paterno, Apellido Materno, Fecha Nacimiento, Direccion,Codigo Postal, Pais, Estado, una casilla de verificacion para indicar si es familiar o no.
- 5) Algunos controles que podrias usar: TextBox, NumberBox, DateTime, ComboBox, CheckBox, Button.
- 6) Ademas, pon un control StatusBar a la forma de altas para indicar cuando has grabado un registro.

El programa debera mostrar la ventana de altas cuando selecciones dicha opcion desde el menu principal de la forma principal. Revisa los ejercicios para ver como usar las propiedades de mostrar y esconder las ventanas.

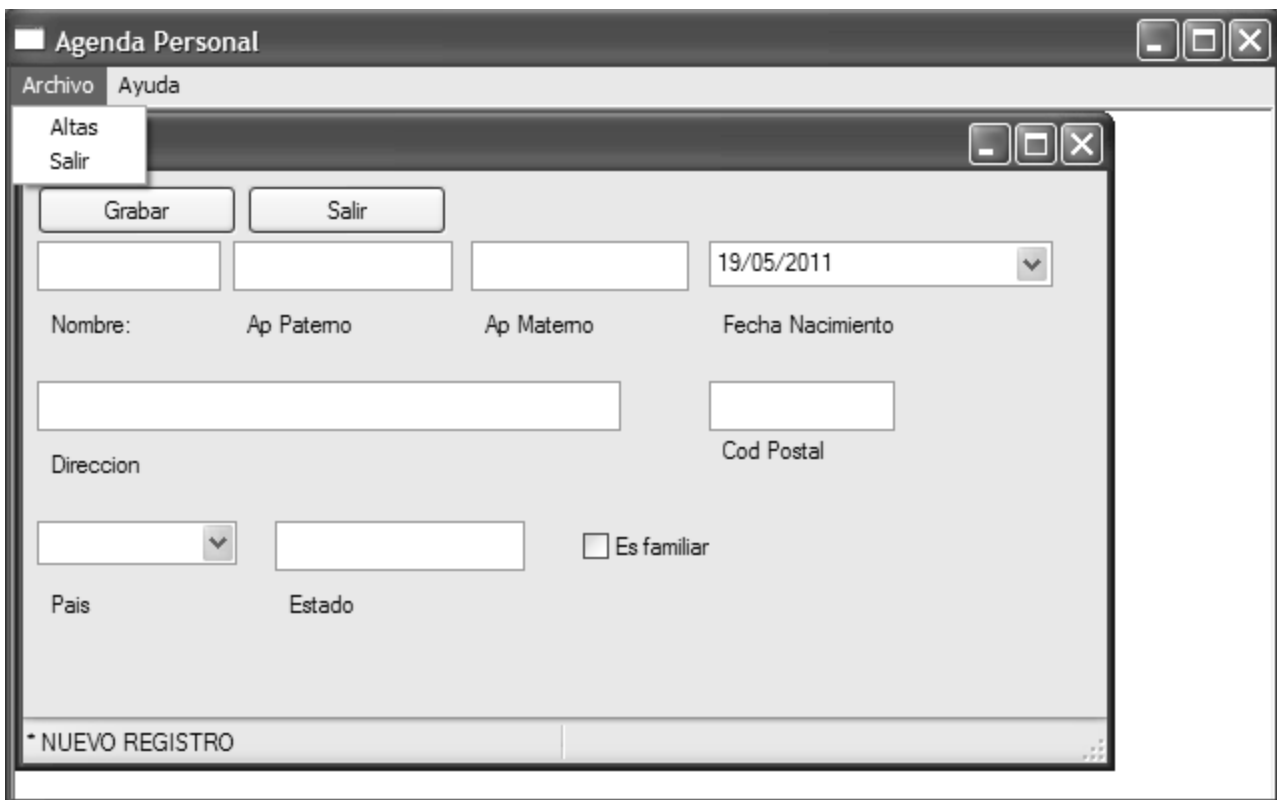
Deberas tener cuidado en crear los controles y agregarlos a las formas correspondientes. Tambien necesitaras crear mas de un procedimiento de forma según el comportamiento del programa.

Revisa los ejercicios referentes a como grabar y leer un archivo para que puedas almacenar la informacion que captures en un archivo de texto llamado AGENDADB.TXT.

Puede que necesites consultar el ejercicio 49 de Arrays Asociativos para manejar una matriz temporal de captura.

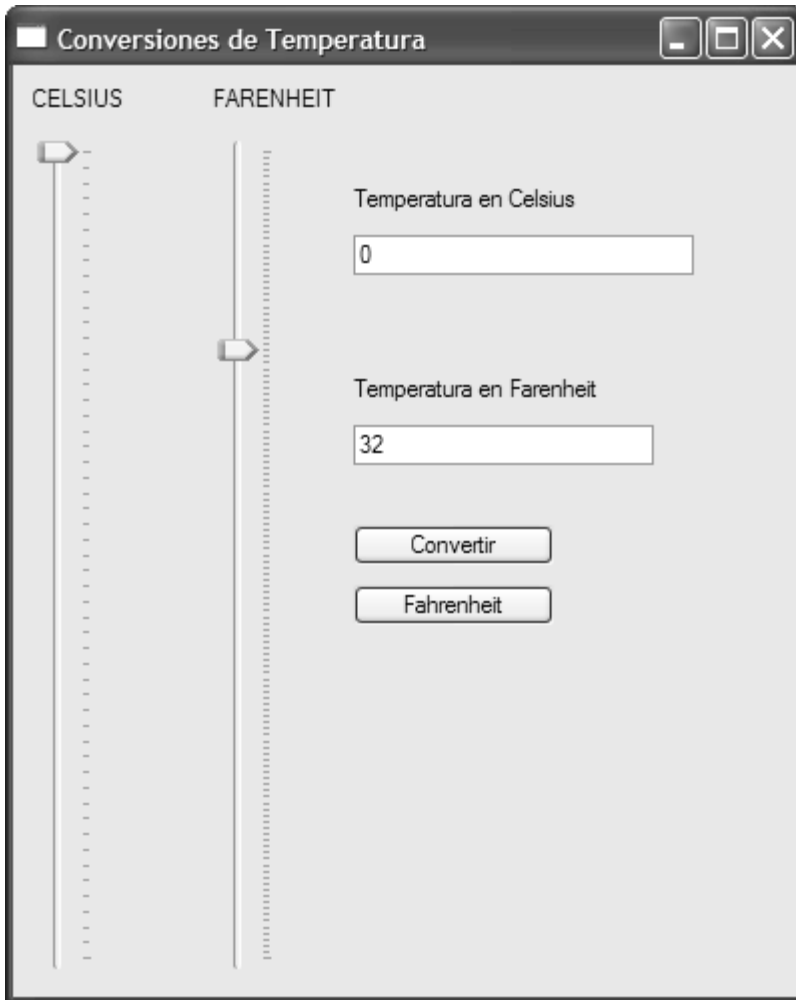
NOTA: Tambien puedes auxiliarte de la herramienta RAD STUDIO APPLICATION para diseñar tu ventana de altas.

Cualquier duda en la creacion de la practica preguntale a tu profesor.



PRACTICA 15

Realiza un programa basado en las practicas 26 y 27 del Cuaderno de Ejercicios de Programacion I, los cuales convierten la temperatura de Celsius a Farenheit y viceversa. Sin embargo, deberas usar solamente el control de Slider para que al mover la temperatura de un grupo, automaticamente se modifique el valor del “otro termometro”.



PRACTICA 16

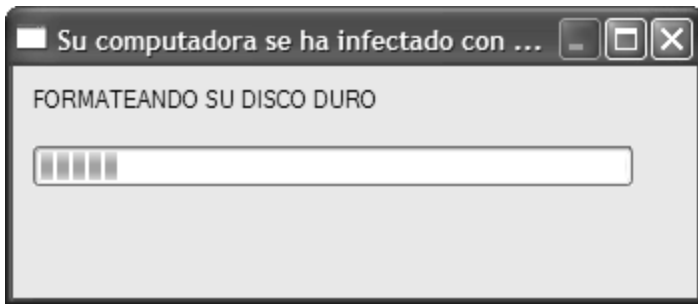
Continuando con el programa de Agenda Telefonica, ahora construye el modulo de Bajas o Eliminacion de datos.

PRACTICA 17

Es tiempo de volver a divertirnos un poco creando otro pseudovirus.

En este caso, haz una ventana que simule simplemente que se esta formateando el disco duro del usuario. No permitas que se cierre la ventana y cuando termine de hacerlo, que envíe un mensaje de error indicando que el disco ha sido formateado o incluso que se reinicie el sistema.

Deberas usar timers, asi como el control ProgressBar y errorBox.



Ejercicio 45 – ScrollBars 2

```
/* ej45_ScrollBars2
Ejemplo de ScrollBars */
#include <radc++.h>

Form    form1("Scroll Bars 2 - RAD C++ Ejemplo");

Label   label("Mover la scrollbar con el raton",-1,100,35,200,20,form1);

//create scrollbars
ScrollBar scroll1(AUTO_ID,100,80,200,25,form1,RCP_HORIZONTAL); //horizontal
ScrollBar scroll2(AUTO_ID,190,120,25,100,form1,RCP_VERTICAL); //vertical

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_SCROLL_CHANGE(scroll1)
        label.caption = str(scroll1.position);

    ON_SCROLL_CHANGE(scroll2)
        label.caption = str(scroll2.position);

    return 0;
}
```

```
rad_main()

    form1.procedure = proc;

    //set scrollbar minimum and maximum range
    scroll1.minRange = -100;
    scroll1.maxRange = 100;
    //you can also use scroll1.setRange(-100,100);

    //NOTE if you do not set range, it is 1 minimum to 100 maximum automatically

    //set scrollbar default position
    scroll1.position = -50;

rad_end()
```

Ejercicio 46 - ToolBar

```
/* ej46_Toolbar
Ejemplo del control Toolbar */
#include <radc++.h>

Form form1("Barras de Herramientas - RAD C++ Ejemplo");
Label label("Toolbar's Opciones de Diseño",-1,100,120,200,20,form1);
CheckBox _flat("Flat toolbar y botones",AUTO_ID,100,150,200,20,form1);
CheckBox _divider("Agregar divisor en lo superior",AUTO_ID,100,180,200,20,form1);

//create toolbar
ToolBar tools(AUTO_ID,form1);
//create toolbar button objects to track clicks
ToolBarItem btn_exit, btn_about;

//create iconlist for toolbar
IconList icl(RCP_SMALL);

//create some icon objects, IDI_XXXXX are predefined in system
Icon icon1(IDI_APPLICATION);
Icon icon2(IDI_EXCLAMATION);
Icon icon3(IDI_HAND);
Icon icon4(IDI_QUESTION);
```

```
FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_COMMAND_BY(btn_exit)
        Application.close();

    ON_COMMAND_BY(btn_about)
        form1.infoBox("RAD C++ Toolbar Ejemplo, Marzo 20, 2007.");

    ON_CHECKBOX_EVENT(_flat)
        tools.flat = _flat.checked;

    ON_CHECKBOX_EVENT(_divider)
        tools.divider = _divider.checked;

    //adjust toolbar when form is resized
    ON_RESIZE()
        tools.adjust();
        //do not use this IF you want toolbar be on absolute position

    return 0;
}

rad_main()

    form1.procedure = proc;

    //add icons to icon list
    icl.add(icon1);
    icl.add(icon2);
    icl.add(icon3);
    icl.add(icon4);

    //attach iconlist with toolbar
    tools.setIconList(icl);
    tools.captions=true;

    //add buttons to toolbar
    tools.add("Item 1",0,AUTO_ID); //second argument is icon id in iconlist
    tools.add("Item 2",1,AUTO_ID);
    btn_exit =tools.add("Salida" ,2,AUTO_ID);
```

```
btn_about=tools.add("Acerca",3,AUTO_ID);
```

```
_flat.checked=true;
```

```
_divider.checked=true;
```

```
rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es una ToolBar o para que sirve?	
¿Cómo agregamos elementos al toolbar?	

Ejercicio 47 - Treeview

```
/*ej47_TreeView
```

```
Ejemplo del control TreeView */
```

```
#include <radc++.h>
```

```
Form form1("Tree View - RAD C++ Ejemplo");
```

```
Label label1("Click algun nodo",-1,50,20,300,20,form1);
```

```
TreeView tree("",AUTO_ID,50,50,300,100,form1); //create treeview
```

```
Label label2("Agregar nuevo nodo bajo el seleccionado",-1,50,170,300,20,form1);
```

```
TextBox text1("Nuevo titulo de nodo",AUTO_ID,50,195,300,20,form1);
```

```
Button btn_add1("Agregar bajo seleccionado",AUTO_ID,50,220,110,25,form1);
```

```
Button btn_add2("Agregar en root",AUTO_ID,170,220,70,25,form1);
```

```
Button btn_rem("Remover seleccionado",AUTO_ID,250,220,100,25,form1);
```

```
TreeNode root,node1,node2;
```

```
FormProcedure proc(FormProcArgs);//prototype of form procedure
```

```
rad_main()
```

```
form1.procedure = proc;
```

```
//NOTE1: 3rd argument of function addNode is the unique Id of node
//      which can later be used to track node.
```

```
//NOTE2: rootNode is predefined, which means no parent node.
```

```
//add two nodes in root
root = tree.addNode(rootNode,"Un Leon", 1);
      tree.addNode(rootNode,"Un Tigre", 2);
```

```
//add node under the node 'root'
node1= tree.addNode(root,"Un Elefante", 3);
//add node under the node 'node1'
node2= tree.addNode(node1,"Un Elefante Azul", 4);
```

```
//expand the root node
root.expand(); //use node.collapse() to close an expanded node
```

```
rad_end()
```

```
FormProcedure proc(FormProcArgs) { //implementation of form procedure
    ON_CLOSE() Application.close();
```

```
    //some node selected
    ON_NODE_CHANGE(tree) {
        label1.caption = tree.selectedNodeText;
        //to track by Id use
        //if(tree.selecteNodeId == 4) //assumed 4 as your node Id
    }
}
```

```
//check specific node if it has been selected, e.g. node1
ON_NODE_SELECT(tree, node1) {
    label1.caption = "'node1' ha sido clickeado";
}
}
```

```
//add new node under selected one (at runtime)
ON_COMMAND_BY(btn_add1) {
    //first get the selected node
    TreeNode tempNode = tree.selectedNode;
    tempNode.addNode(text1.text,AUTO_ID);
    tempNode.expand();
}
}
```

```
//add new node in root
```

```

ON_COMMAND_BY(btn_add2) {
    root = tree.addNode(rootNode,text1.text,AUTO_ID);
}

//remove the selected node
ON_COMMAND_BY(btn_rem) {
    if(tree.total == 0) return 0; //no node in treeview
    //first get the selected node
    TreeNode tempNode = tree.selectedNode;
    //now remove it
    tempNode.remove();
}
return 0;
}

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un Treeview?	
¿Cómo agregamos elementos al control?	

Ejercicio 48 – RichTextBox Intermedio

```

/* ej48_RichTextBoxIntermedio
Uso mas a fondo del control RichTextBox */
#include <radc++.h>

Form    form1("Rich Text Box Intermedio - RAD C++ Ejemplo");

COLORREF last_color=0x000000; //some default color - black

Button   bbtn("Negrita", AUTO_ID,50,10,50,20,form1);
Button   ibtn("Cursiva",AUTO_ID,100,10,50,20,form1);
Button   ubtn("Subrayado",AUTO_ID,150,10,70,20,form1);
ColorButton cbtn(AUTO_ID,220,10,50,20,form1,last_color);

Label label("Margenes", -1,50,40,60,20,form1);

```

```
Track track(AUTO_ID,110,40,200,20,form1);

RichTextBox rich("", AUTO_ID,50,70,300,180,form1);

FormProcedure proc(FormProcArgs) { //implementation of form procedure
    ON_CLOSE() Application.close();

    //bold the selection
    ON_COMMAND_BY(bbtn) {
        rich.selectionToBold();
        rich.focus();
    }
    //make the selected text italic
    ON_COMMAND_BY(ibtn) {
        rich.selectionToItalic();
        rich.focus();
    }
    //underline selected text
    ON_COMMAND_BY(ubtn) {
        rich.selectionToUnderline();
        rich.focus();
    }

    //change color of selected text
    ON_COMMAND_BY(cbtn) {
        if(form1.selectColor(last_color)) {
            rich.colorSelection(last_color);
            cbtn.color = last_color;
        }
        rich.focus();
    }
    //change the left and right margins of richtext box
    ON_TRACK_CHANGE(track) {
        rich.leftMargin = track.position;
        rich.rightMargin = track.position;
    }

    return 0;
}

rad_main()
```

```
form1.procedure = proc;
track.maxRange = 50; //set range of trackbar

rad_end()
```

Ejercicio 49 – Arrays Asociativos

```
/* ej49_Arrays
Ejemplo de Arreglos asociativos */
#include <radc++.h>

Form          form1("Arrays Asociativos - RAD C++ Ejemplo");
Label  label1("EMPLEADO BIO\n\n",-1,50,50,150,200,form1);
SunkLine  line1(210,50,2,200,form1);
Label  label2("EMPLEADO INFO\n\n",-1,250,50,100,200,form1);

//create an associative array to hold string values
createAssocArray(String, bio_data);

//create an associative array to hold integers
createAssocArray(int, extra_info);

//NOTE: function createAssocArray accepts two arguments, where 1st is type of array elements,
//      and second is the name of variable.

rad_main()

//add some values to array of strings 'bio_data'
bio_data["nombre"]          = "Ali Imran";
bio_data["titulo"]          = "Programmer";
bio_data["hobbies"]         = "Programming";
bio_data["NIC#"]            = "PM-8500881";
bio_data["ciudad"]          = "Islamabad";
bio_data["pais"]            = "Pakistan";

//add some values to array of integers 'extra_info'
extra_info["edad"]          = 30;
extra_info["estatura"]      = 6.1;
```

```

extra_info["salario"]    = 30000;
extra_info["grado"]      = 19;

for(int i=0; i<bio_data.length; i++) {
    label1.caption = label1.caption + "\n" + bio_data.indexName(i) + "\t" + bio_data[i];
}

for(int i=0; i<extra_info.length; i++) {
    label2.caption = label2.caption + "\n" + extra_info.indexName(i) + "\t" + str(extra_info[i]);
}

//to accesss in field separately use
//String test = bio_data["country"];
//and for integer as we have in above example
//int test = extra_info["salary"];

rad_end()

```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué son los Arrays Asociativos?	
¿Cómo creamos un array asociativo?	

Ejercicio 50 – Grupos de Objetos

```

/* ej50_GruposObjetosLogicos
Ejemplo con varios controles agrupados */
#include <radc++.h>

Form    form1("Grupo de Objetos (agrupando objetos) - RAD C++ Ejemplo");

//5 Object that will make a group, trying different controls
Button  obj1("obj1",AUTO_ID,50,100,100,20,form1);
TextBox obj2("obj2",AUTO_ID,50,130,100,20,form1);
ScrollBar obj3(    AUTO_ID,50,160,100,20,form1);
CheckBox obj4("obj4",AUTO_ID,50,190,100,20,form1);

```

```
//a separate form to be part of this logical group
Form obj5("obj5",200,200,100,50,RCP_POPUP,true,true,false,false,false,form1);

ObjectGroup my_group; //creat a logicla group variable

Label label("Grupo logico siguiendo objetos",-1,40,50,150,30,form1);
Separator line (200,0,2,300,form1);

//buttons to perform functions of logical group
Button btn_hide("Ocultar Grupo",AUTO_ID,250,50,100,25,form1);
Button btn_show("Mostrar Grupo",AUTO_ID,250,80,100,25,form1);
Button btn_disa("Desactivar Grupo",AUTO_ID,250,110,100,25,form1);
Button btn_enab("Activar Grupo",AUTO_ID,250,140,100,25,form1);

FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_COMMAND_BY(btn_hide)
        my_group.hide(); //hide the group objects

    ON_COMMAND_BY(btn_show)
        my_group.show(); //show all group objects, if any or all were hidden

    ON_COMMAND_BY(btn_disa)
        my_group.disable(); //disable all gorup objects

    ON_COMMAND_BY(btn_enab)
        my_group.enable(); //enable all group objects, if any or all were duisabled

    return 0;
}

rad_main()

form1.procedure = proc;

//add objects to group, either form, button, check, radio, or any kind of object
```

```

my_group.add(obj1);
my_group.add(obj2);
my_group.add(obj3);
my_group.add(obj4);
my_group.add(obj5);

//reposition child for obj5
obj5.left = form1.left + 50;
obj5.top = form1.top + form1.height - 60;
obj5.icon = Icon::Icon(IDI_EXCLAMATION);

//set focus on main form
form1.focus();

```

```
rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Para que nos sirve agrupar objetos?	
¿Cómo creamos un grupo de objetos?	
¿Cómo asociamos los objetos al grupo?	

Ejercicio 51 - CoolBars

```

/* ej51_CoolBars
Ejemplo de varios controles colocados en barras de herramientas */
#include <radc++.h>

Form form1("Cool Bars / Rebar - RAD C++ Ejemplo");

//create three controls that we will add to the coolbar
//NOTE : controls will report to form on events so parent of controls must be form1
ComboBox combo("Soy un combo", AUTO_ID,0,0,70,100,form1);
CheckBox check("Soy un check", AUTO_ID,0,0,100,25,form1);
Button btn("Soy un boton",AUTO_ID,0,0,120,25,form1);
Radio radio("Soy un radio", AUTO_ID,0,0, 70,25,form1);

```

```
//create a coolbar
CoolBar cool(AUTO_ID,form1);

//main form procedure
FormProcedure proc(FormProcArgs) {
    ON_CLOSE() Application.close();

    //adjust coolbar properly on form, when form is resized
    ON_RESIZE()
        cool.adjust();

    return 0;
}

rad_main()

    form1.procedure = proc;

    //add bands to coolbar
    cool.add("",combo);
    cool.add("",check);

    //add band on new line
    cool.add("",btn, RCP_NEWBAND);

    //add band with custom caption 'Extra' with text color blue, color in hex 0xFF0000
    cool.add("Extra",radio,RCP_SAME,0xFF0000);
    //RCP_SAME means add new band on same line

    //add some values to combo
    combo.add("Item1");
    combo.add("Item2");
    combo.select(0); //select first item in combo

rad_end()
```

Responde a lo siguiente:

Pregunta	Respuesta
¿Qué es un CoolBar?	
¿Cómo agregamos los controles a la banda?	

Ejercicio 52 – Toolbar y CoolBar

```
/* ej52_ToolBarIncrustada
Ejemplo de como poner una toolbar junto con Coolbar */
#include <radc++.h>

Form form1("Cool Bar con Toolbar - RAD C++ Ejemplo");
//create a coolbar
CoolBar cool(AUTO_ID,form1);

/*
    NOTE:
    If you need to add a toolbar to a coolbar control, you will have to follow:
    1.first we create a new form with no style
    2.then create a toolbar and add to the newly created form
    3.then add this new form to coolbar
*/
//Create another form with no style, we will add toolbar to this form
Form form2("",0,0,200,40,RCP_NONE,true,true,false,false,false,form1,true);
//create toolbar as child of form2 which has no borders and style
ToolBar tools(AUTO_ID,form2);
//create toolbar button objects to track clicks
ToolBarItem btn_exit, btn_about;

//one more object for coolbar
Track trk(AUTO_ID,0,0,100,20,form1,30);

//main form procedure
FormProcedure mproc(FormProcArgs);
//procedure for the child form containing toolbar
FormProcedure cproc(FormProcArgs);

rad_main()

    form1.procedure = mproc;
    form2.procedure = cproc;

    //create iconlist for toolbar
    IconList icl(RCP_SMALL);
    //add some icons to icon list
    icl.add(Icon::Icon(IDI_APPLICATION));
```

```
icl.add(Icon::Icon(IDI_EXCLAMATION));
icl.add(Icon::Icon(IDI_HAND));
icl.add(Icon::Icon(IDI_QUESTION));

//attach iconlist with toolbar
tools.setIconList(icl);

//show text with icons
tools.captions=true;
//don't push icons to next line when not enough space
tools.wrapable=false; //test compile after removing this line and decrease form width

//add buttons to toolbar
tools.add("Item 1",0,AUTO_ID); //second argument is icon id in iconlist
tools.add("Item 2",1,AUTO_ID);
btn_exit =tools.add("Salida" ,2,AUTO_ID);
btn_about=tools.add("Acerca" ,3,AUTO_ID);

//add the form2 which had not style, to coolbar so that toolabr remains in place properly
cool.add("",form2);
//add track to coolbar
cool.add("",trk);

rad_end()

//implementation of forms' procedures
FormProcedure cproc(FormProcArgs) {

    //forward toolbar commands to form1 so that it can take action
    FORWARD_COMMANDS(form1);

    //adjust toolbar properly when form is resized
    ON_RESIZE()
        tools.adjust();

    return 0;
}
FormProcedure mproc(FormProcArgs) {
    ON_CLOSE() Application.close();

    ON_COMMAND_BY(btn_exit)
        Application.close();
}
```

```
ON_COMMAND_BY(btn_about)
    form1.infoBox("RAD C++ Coolbar y Toolbar Ejemplo.");

//adjust coolbar properly when form is resized
ON_RESIZE()
    cool.adjust();

return 0;
}
```

PRACTICA 18

Nuevamente continua con el programa de Agenda, y construye ahora un modulo para Cambios o modificacion de datos.

PRACTICA 19

Y finalmente, construye el programa de Agenda con los modulos creados anteriormente en un unico programa. Ademas, deberas completar el menu principal en la parte superior y crear una barra de herramientas para que sea mas facil algunas de las opciones.

PRACTICA 20

Desarrolla un programa que permita jugar EL AHORCADO de manera visual. Podras basarte en parte del codigo del Cuaderno de Ejercicios Programaicon I o en algun otro. Aquí se utiliza mucho la librería <string.h> para manejo de cadenas.

Pero, deberas construir de manera visual al hombre y las partes que lo componen hasta que se ahorque si pierdes.