

LISTADO DEL PROGRAMA COMANDO 1,9

IMPORTANTE: NO INTENTE COPIAR Y PEGAR INSTRUCCIONES TOMADAS DE ESTE LISTADO SIN CONOCER SU FUNCIÓN Y SU LUGAR ESPECIFICO
EL LISTADO DEL PROGRAMA ES SOLO PARA TERMINAR DE COMPRENDER EL MISMO.

RECUERDE QUE // ES COMENTARIO, SI UNA LINEA DEL PROGRAMA COMIENZA CON // NO ES UNA INSTRUCCIÓN A EJECUTARSE SINO UN COMENTARIO. EL PROGRAMADOR PUEDE HABER CONVERTIDO CIERTAS INSTRUCCIONES EN COMENTARIOS EN LUGAR DE ELEIMINARLAS CON EL FIN DE PRESERVAR LAS MISMAS, CON OBJETO DE RECORDAR SU RAZOMIENTO. SI UD DECIDE BORRARLAS NO GENERA DIFERENCIA EN LA EJECUCIÓN DEL PROGRAMA.

```
//-----  
#include <vcl.h>  
#pragma hdrstop  
  
#include "Unit1.h"  
//-----  
#pragma package(smart_init)  
#pragma link "LPComponent"  
#pragma link "LPDrawLayers"  
#pragma link "LPTransparentControl"  
#pragma link "SLComponentCollection"  
#pragma link "ULBasicControl"  
#pragma link "ULImage"  
#pragma link "ALAudioOut"  
#pragma link "ALBasicAudioOut"  
#pragma link "ALCommonPlayer"  
#pragma link "ALWavePlayer"  
#pragma link "ILSegmentIndicator"  
#pragma link "ILSegmentText"  
#pragma resource "*.dfm"  
TForm1 *Form1;  
//Defino aqui variables globales que usare en todo el programa  
  
int balas=50,cargadores=4,granadas=4,vidas=4,numblanco;  
int blanco1=0,blanco2=0,blanco3=0,dispara1=0,dispara2=0,dispara3=0;  
int enemigos=0,amigos=0,puntaje=0;  
  
//-----  
__fastcall TForm1::TForm1(TComponent* Owner)  
    : TForm(Owner)  
{  
}  
//-----  
  
void __fastcall TForm1::Image5MouseMove(TObject *Sender, TShiftState Shift,  
    int X, int Y)  
{  
  
    ULImage2->Visible=false;  
    ULImage1->Visible=false;  
    ULImage1->Visible=true;  
  
}  
//-----  
void __fastcall TForm1::Image7MouseMove(TObject *Sender, TShiftState Shift,  
    int X, int Y)  
{  
  
    // ULImage1->Visible=false;  
    // ULImage2->Visible=false;
```

```

// ULImage2->Visible=true;
}
//-----
void __fastcall TForm1::Image1MouseMove(TObject *Sender, TShiftState Shift,
    int X, int Y)
{
    ULImage2->Visible=false;
    ULImage3->Visible=false;

    //ULImage1->Visible=false;
    // Image1->Visible=true;//reafirmo imagen puerta o equivalente

    ULImage1->Visible=true;
    ULImage1->Transparent=true;
}
//-----
void __fastcall TForm1::Image3MouseMove(TObject *Sender, TShiftState Shift,
    int X, int Y)
{
    ULImage1->Visible=false;
    ULImage3->Visible=false;

    //ULImage2->Visible=false;

    ULImage2->Visible=true;
    ULImage2->Transparent=true;
}
//-----
void __fastcall TForm1::FormCreate(TObject *Sender)
{
    ULImage1->Visible=false;
    ULImage3->Visible=false;

    ULImage2->Visible=false;
}
//-----
void __fastcall TForm1::Image2MouseMove(TObject *Sender, TShiftState Shift,
    int X, int Y)
{
    ULImage1->Visible=false;
    ULImage2->Visible=false;

    ULImage3->Visible=true;
    ULImage3->Transparent=true;
}
//-----
void __fastcall TForm1::Timer1Timer(TObject *Sender)
{
    //Luego de este tiempo de timer se hace nuevamente invisible la imagen de la explosion
    ULImage4->Visible=false;//Explosion 1 invisible
    //dispara1=0;// informa que se acabo el lapso del disparo1
    Timer1->Enabled=false; //Apaga timer
}

```

```

}
//-----

void __fastcall TForm1::Image1Click(TObject *Sender)
{

    if(balas>0)

        {

            //instrucciones se ejecutan si condición se cumple
            // Image1->Visible=true;//reafirmo imagen puerta o equivalente

            UImage4->Visible=true;//Explosion 1 visible

            ALWavePlayer1->Start(); //disparo del arma

            dispara1=1;// con el 1 indica que se ha disparado en puerta 1

            //descuenta balas
            balas= balas-5;

            if(balas>0)

                {

                    //instrucciones se ejecutan si condición se cumple
                    ILSegmentText1->Text=balas;
                }

            else
                {

                    //instrucciones se ejecutan si condición NO se cumple
                    ILSegmentText1->Text="00";
                }

            Timer1->Enabled=true;//Habilita Timer que luego hace invisible explosion1

        }

    else
        {

            //instrucciones se ejecutan si condición NO se cumple
            //debe recargar
            Label5->Visible=true;
            //ALWavePlayer2->Enabled=true;
            ALWavePlayer2->Start();

        }

}

```

```
//-----
```

```
void __fastcall TForm1::Image2Click(TObject *Sender)  
{
```

```
    if(balas>0)
```

```
    {
```

```
        //instrucciones se ejecutan si condición se cumple
```

```
        UImage5->Visible=true;//Explosion 2 visible  
        ALWavePlayer1->Start(); //disparo del arma  
        dispara2=1;// con el 1 indica que se ha disparado en puerta 2
```

```
        //descuenta balas
```

```
        balas= balas-5;
```

```
            if(balas>0)
```

```
            {
```

```
                //instrucciones se ejecutan si condición se cumple
```

```
                ILSegmentText1->Text=balas;
```

```
            }
```

```
        else
```

```
        {
```

```
            //instrucciones se ejecutan si condición NO se cumple
```

```
            ILSegmentText1->Text="00";
```

```
        }
```

```
        Timer2->Enabled=true;//Habilita Timer que luego hace invisible explosion2
```

```
    }
```

```
else
```

```
{
```

```
    //instrucciones se ejecutan si condición NO se cumple
```

```
    //debe recargar
```

```
    Label5->Visible=true;
```

```
    //ALWavePlayer2->Enabled=true;
```

```
    ALWavePlayer2->Start();
```

```
}
```

```
}
```

```
//-----
```

```
void __fastcall TForm1::Image3Click(TObject *Sender)
```

```
{
```

```

if(balas>0)
{
//instrucciones se ejecutan si condición se cumple

UIImage6->Visible=true;//Explosion 3 visible
ALWavePlayer1->Start(); //disparo del arma

dispara3=1;// con el 1 indica que se ha disparado en puerta 3

//descuenta balas
balas= balas-5;

    if(balas>0)
        {
            //instrucciones se ejecutan si condición se cumple
            ILSegmentText1->Text=balas;
        }
    else
        {
            //instrucciones se ejecutan si condición NO se cumple
            ILSegmentText1->Text="00";
        }

    Timer3->Enabled=true;//Habilita Timer que luego hace invisible explosion 3
}

else
{
//instrucciones se ejecutan si condición NO se cumple
//debe recargar
Label5->Visible=true;
//ALWavePlayer2->Enabled=true;
ALWavePlayer2->Start();
}
}
//-----

void __fastcall TForm1::Timer2Timer(TObject *Sender)
{
//Luego de este tiempo de timer se hace nuevamente invisible la imagen de la explosion
UIImage5->Visible=false;//Explosion 2 invisible
//dispara2=0;// informa que se acabo el lapso del disparo2
Timer2->Enabled=false; //Apaga timer
}
//-----

void __fastcall TForm1::Timer3Timer(TObject *Sender)
{
//Luego de este tiempo de timer se hace nuevamente invisible la imagen de la explosion
UIImage6->Visible=false;//Explosion 3 invisible
Timer3->Enabled=false; //Apaga timer
}

```

```

}
//-----

void __fastcall TForm1::ULImage7Click(TObject *Sender)
{
    balas=50;//restaurar contador de balas
    ALWavePlayer2->Start(); //reproduce de recarga
    ULImage7->Visible=false;//oculta cargador 1
    Label5->Visible=false; //oculta cartel de recargar
    ILSegmentText1->Text=balas;//coloca visor de balas a maximo 50

}
//-----

void __fastcall TForm1::ULImage19Click(TObject *Sender)
{
    //Al presionar el lanzagranadas

if(granadas>0) //dispara granadas si hay

    {

        //instrucciones se ejecutan si condición se cumple

        Timer9->Enabled=false; //detengo timer control aciertos ventana1
        Timer11->Enabled=false; //detengo timer control aciertos ventana2
        Timer13->Enabled=false; //detengo timer control aciertos ventana3

        dispara1=1;// simula disparo a ventana 1 al tirar granada
        dispara2=1;// simula disparo a ventana 2 al tirar granada
        dispara3=1;// simula disparo a ventana 3 al tirar granada

        Timer9->Enabled=true; // inicia timer de control de aciertos y desaciertos ventana

        Timer11->Enabled=true; // inicia timer de control de aciertos y desaciertos ventana 2

        Timer13->Enabled=true; // inicia timer de control de aciertos y desaciertos ventana 3

        ULImage24->Visible=true;
        ALWavePlayer3->Start();
        granadas=granadas-1; //descuenta granadas
    }

else
    {

        //instrucciones se ejecutan si condición NO se cumple

    }

//control de granadas

switch (granadas) //Borra granadas usadas
{
    case 0:// borra granada 1
        ULImage15->Visible=false;
        break;

```

```

case 1:// borra granada 2
    ULImage16->Visible=false;
break;
case 2:// borra granada 3
    ULImage17->Visible=false;
break;

case 3:// borra granada 4
    ULImage18->Visible=false;
break;

}

```

```

Timer4->Enabled=true;

```

```

}
//-----

```

```

void __fastcall TForm1::Timer4Timer(TObject *Sender)
{
    //Tiempo del lanzagranadas
    // dispara1=0;// informa que se acabo el lapso del simula disparo para granada
    ULImage24->Visible=false;
    Timer4->Enabled=false;
}
//-----

```

```

void __fastcall TForm1::ULImage11Click(TObject *Sender)
{
    //carga cuatro granadas inferiores
    ULImage15->Visible=true;
    ULImage16->Visible=true;
    ULImage17->Visible=true;
    ULImage18->Visible=true;
    granadas=4;//restaura contador granadas

```

```

    ULImage11->Visible=false; // Invisible una carga de granadas
}
//-----

```

```

void __fastcall TForm1::ULImage12Click(TObject *Sender)
{
    //carga cuatro granadas inferiores
    ULImage15->Visible=true;
    ULImage16->Visible=true;
    ULImage17->Visible=true;
    ULImage18->Visible=true;
    granadas=4;//restaura contador granadas

```

```

    ULImage12->Visible=false; // Invisible una carga de granadas
}
//-----

```

```

void __fastcall TForm1::ULImage13Click(TObject *Sender)

```

```

{

//carga cuatro granadas inferiores
UIImage15->Visible=true;
UIImage16->Visible=true;
UIImage17->Visible=true;
UIImage18->Visible=true;
granadas=4;//restaura contador granadas

UIImage13->Visible=false; // Invisible una carga de granadas

}
//-----

void __fastcall TForm1::UIImage14Click(TObject *Sender)
{

//carga cuatro granadas inferiores
UIImage15->Visible=true;
UIImage16->Visible=true;
UIImage17->Visible=true;
UIImage18->Visible=true;
granadas=4;//restaura contador granadas

UIImage14->Visible=false; // Invisible una carga de granadas

}
//-----

void __fastcall TForm1::UIImage8Click(TObject *Sender)
{

balas=50;//restaurar contador de balas
ALWavePlayer2->Start(); //reproduce de recarga
UIImage8->Visible=false;//oculta cargador 2
Label5->Visible=false; //oculta cartel de recargar
ILSegmentText1->Text=balas;//coloca visor de balas a maximo 50

}
//-----

void __fastcall TForm1::UIImage9Click(TObject *Sender)
{

balas=50;//restaurar contador de balas
ALWavePlayer2->Start(); //reproduce de recarga
UIImage9->Visible=false;//oculta cargador 3
Label5->Visible=false; //oculta cartel de recargar
ILSegmentText1->Text=balas;//coloca visor de balas a maximo 50

}
//-----

```



```

void __fastcall TForm1::ULImage10Click(TObject *Sender)
{
    balas=50;//restaurar contador de balas
    ALWavePlayer2->Start(); //reproduce de recarga
    ULImage10->Visible=false;//oculta cargador 3
    Label5->Visible=false; //oculta cartel de recargar
    ILSegmentText1->Text=balas;//coloca visor de balas a maximo 50

}
//-----

```

```

void __fastcall TForm1::Timer6Timer(TObject *Sender)
{
    //Timer5->Enabled=false;

    int rand(void);
    void srand(unsigned int semilla);
    srand( (unsigned int)time( NULL ) );

    //La última instrucción toma el tiempo como semilla.
    //de este modo no comienza la generación con el mismo número.

    //*****
    //Luego utilizamos la función RAND donde lo deseamos.

    numblanco = rand()%14+1;

    //El valor 14 +1 da el rango maximo de los
    //numeros aleatorios que apareceran, en este caso de 1 a 14

    Timer5->Enabled=true;

    Edit5->Text=numblanco;

    //Timer6->Enabled=false;

}
//-----

```

```

void __fastcall TForm1::Timer5Timer(TObject *Sender)
{
    //Timer que inicia el cambio de la imagen en puerta 2

    if ( ULImage2->Visible==false)
    {
        switch (numblanco)
        {
            case 1:// imagen de
                Image2->Picture->LoadFromFile("puerta_abierta.jpg");
                blanco2=2;//no es ni amigo ni enemigo
                break;

            case 2:// imagen de
                Image2->Picture->LoadFromFile("HomeroApunt.jpg");
                blanco2=1;//enemigo

```

```
break;
case 3:// imagen de
    Image2->Picture->LoadFromFile("zombi1.jpg");
    blanco2=1;//enemigo
break;

case 4:// imagen de

    Image2->Picture->LoadFromFile("bebe1.jpg");
    blanco2=0;//amigo
break;

case 5:// imagen de

    Image2->Picture->LoadFromFile("Gerrillero1.jpg");
    blanco2=1;//enemigo
break;

case 6:// imagen de

    Image2->Picture->LoadFromFile("puerta_abierta.jpg");
    blanco2=2;// ni amigo ni enemigo
break;
case 7:// imagen de

    Image2->Picture->LoadFromFile("BartApunta.jpg");
    blanco2=1;// enemigo
break;

case 8:// imagen de
    Image2->Picture->LoadFromFile("Gerrillero2.jpg");
    blanco2=1;//enemigo
break;

case 9:// imagen de
    Image2->Picture->LoadFromFile("cajaarmas.jpg");
    blanco2=3;// caja con armas

break;

case 10:// imagen de
    Image2->Picture->LoadFromFile("Soldado1.jpg");
    blanco2=1;//enemigo
break;

    case 11:// imagen de
    Image2->Picture->LoadFromFile("Senoranino4.jpg");
    blanco2=0;// amigo

break;

    case 12:// imagen de
    Image2->Picture->LoadFromFile("zombiArmado2.jpg");
    blanco2=1;// enemigo
break;

    case 13:// imagen de
    Image2->Picture->LoadFromFile("zombiArmado.jpg");
    blanco2=1;//enemigo
```

```

break;

    case 14:// imagen de
    Image2->Picture->LoadFromFile("senoranino2.jpg");
    blanco2=0;// amigo

break;

}

Timer11->Enabled=true; // inicia timer de control de aciertos y desaciertos

} //fin del if

}
//-----

void __fastcall TForm1::Timer7Timer(TObject *Sender)
{
    //Timer que inicia el cambio de la imagen en puerta 1

if ( UImage1->Visible==false)
{

switch (numblanco)
{
case 1:// imagen de
    Image1->Picture->LoadFromFile("puerta_abierta.jpg");
    blanco1=2;//no es ni amigo ni enemigo
break;

case 2:// imagen de
    Image1->Picture->LoadFromFile("HomeroApunt.jpg");
    blanco1=1;//enemigo
break;
case 3:// imagen de
    Image1->Picture->LoadFromFile("zombi1.jpg");
    blanco1=1;//enemigo
break;

case 4:// imagen de

    Image1->Picture->LoadFromFile("bebe1.jpg");
    blanco1=0;//amigo
break;

case 5:// imagen de

    Image1->Picture->LoadFromFile("Gerrillero1.jpg");
    blanco1=1;//enemigo
break;

case 6:// imagen de

    Image1->Picture->LoadFromFile("puerta_abierta.jpg");
    blanco1=2;// ni amigo ni enemigo
break;
case 7:// imagen de

    Image1->Picture->LoadFromFile("BartApunta.jpg");

```

```

        blanco1=1;// enemigo
break;

case 8:// imagen de
    Image1->Picture->LoadFromFile("Gerrillero2.jpg");
    blanco1=1;//enemigo
break;

case 9:// imagen de
    Image1->Picture->LoadFromFile("puerta_abierta.jpg");
    blanco1=2;// ni amigo ni enemigo

break;

case 10:// imagen de
    Image1->Picture->LoadFromFile("Soldado1.jpg");
    blanco1=1;//enemigo
break;

    case 11:// imagen de
    Image1->Picture->LoadFromFile("Senoranino4.jpg");
    blanco1=0;// amigo

break;

    case 12:// imagen de
    Image1->Picture->LoadFromFile("zombiArmado2.jpg");
    blanco1=1;// enemigo
break;

    case 13:// imagen de
    Image1->Picture->LoadFromFile("zombiArmado.jpg");
    blanco1=1;//enemigo

break;

    case 14:// imagen de
    Image1->Picture->LoadFromFile("dama1.jpg");
    blanco1=0;// amigo

break;

}

Timer9->Enabled=true; // inicia timer de control de aciertos y desaciertos

}

}

//-----

void __fastcall TForm1::Timer8Timer(TObject *Sender)
{
    //Este Timer permite que las puertas inicialmente cerradas se abran. Mediante un cambio de imagenes
    Image1->Picture->LoadFromFile("puerta_abierta.jpg");//Puerta izquierda

```

```
Image2->Picture->LoadFromFile("puerta_abierta.jpg");//Puerta del medio
Image3->Picture->LoadFromFile("puerta_abierta.jpg");//Puerta derecha
ULImage25->Visible=true;//hace visible imagen rifle de arcenal
Timer6->Enabled=true;//habilita Timer que genera aleatoriamente imagenes de blancos
```

```
Timer8->Enabled=false;
```

```
}
//-----
```

```
void __fastcall TForm1::ULImage25Click(TObject *Sender)
```

```
{
    Image1->Enabled=true;
    Image2->Enabled=true;
    Image3->Enabled=true;

    ULImage25->Visible=false;
}
```

```
//-----
```

```
void __fastcall TForm1::Timer9Timer(TObject *Sender)
```

```
{

//timer de control de aciertos y desaciertos ventana 1

if((blanco1==1)&& (dispara1==1))//blanco enemigo y disparo efectuado

{

//instrucciones se ejecutan si condición se cumple
enemigos = enemigos+1;//sumamos enemigos
puntaje=puntaje+300;//sumamos 300 puntos
Edit1->Text=enemigos; //Muestra en Edit1 enemigos caidos
Edit3->Text=puntaje; //Muestra en Edit3 puntaje actual

Image1->Picture->LoadFromFile("sangre1.jpg");
blanco1=2;//no es ni amigo ni enemigo -cambia status de blanco
ALWavePlayer4->Start(); //grito de otro

} //fin del if blanco enemigo y disparo efectuado
```

```
if((blanco1==0)&& (dispara1==1)) //blanco amigo y disparo efectuado
```

```
{

//instrucciones se ejecutan si condición se cumple
amigos = amigos+1;//sumamos amigos
puntaje=puntaje-900;//restamos 900 puntos
Edit2->Text=amigos; //Muestra en Edit2 amigos caidos
Edit3->Text=puntaje; //Muestra en Edit3 puntaje actual

Image1->Picture->LoadFromFile("sangre1.jpg");
ALWavePlayer6->Start(); //grito de amigo

blanco1=2;//no es ni amigo ni enemigo -cambia status de blanco

} //fin del if blanco amigo y disparo efectuado
```

```

if((blanco1==1)&& (dispara1==0)) //pierdo vidas

{

//instrucciones se ejecutan si condición se cumple
// ALWavePlayer6->Start(); // Disparo del enemigo
vidas=vidas-1;//restamos una vida

//control de vidas

switch (vidas) //Borra vidas perdidas
{

case 0:// borra vida 1
    ULImage23->Visible=false;
    ALWavePlayer5->Start(); //grito soldado
    Timer10->Enabled=true; //inicia Timer 10 muerte del combatiente

break;

case 1:// borra vida 2
    ULImage22->Visible=false;
    ALWavePlayer5->Start(); //grito soldado

break;

case 2:// borra vida 3
    ULImage21->Visible=false;
    ALWavePlayer5->Start(); //grito soldado
break;

case 3:// borra vida4
    ULImage20->Visible=false;
    ALWavePlayer5->Start(); //grito soldado
break;

} //fin del switch-case de borrar vidas perdidas

} //fin del if pierdo vidas

dispara1=0;//fin de disparo1 util para rifle y granada

Timer9->Enabled=false;// apago timer de control de aciertos y desaciertos
}
//-----

void __fastcall TForm1::Timer10Timer(TObject *Sender)
{
    //Muerte o victoria del combatiente
    // Resumen de reultados
    Memo1->Lines->Clear();//limpia memo
    Memo1->Lines->Add("-----");
    Memo1->Lines->Add("");
    Memo1->Lines->Add("Rango="+Edit4->Text);
    Memo1->Lines->Add("");

if(puntaje>=30000)

{

```

```

//instrucciones se ejecutan si condición se cumple
Memo1->Lines->Add("Status : Retirado con honores.");

Memo1->Lines->Add("");
Memo1->Lines->Add("-----");
Memo1->Lines->Add("");
Memo1->Lines->Add("Puntaje="+Edit3->Text);
Memo1->Lines->Add("");
Memo1->Lines->Add("Enemigos Eliminados =" +Edit1->Text);
Memo1->Lines->Add("");
Memo1->Lines->Add("Amigos Eliminados= " +Edit2->Text);
Memo1->Lines->Add("");
Memo1->Lines->Add("-----");
Memo1->Lines->SaveToFile("resumen.txt");

ShellExecute(NULL,"open","retiro.exe",NULL,"",SW_SHOWNORMAL);

Application->Terminate();

}

else
{

//instrucciones se ejecutan si condición NO se cumple

Memo1->Lines->Add("Status : Muerto en acción.");
Memo1->Lines->Add("");
Memo1->Lines->Add("-----");
Memo1->Lines->Add("");
Memo1->Lines->Add("Puntaje="+Edit3->Text);
Memo1->Lines->Add("");
Memo1->Lines->Add("Enemigos Eliminados =" +Edit1->Text);
Memo1->Lines->Add("");
Memo1->Lines->Add("Amigos Eliminados= " +Edit2->Text);
Memo1->Lines->Add("");
Memo1->Lines->Add("-----");
Memo1->Lines->SaveToFile("resumen.txt");
ShellExecute(NULL,"open","FinJuego.exe",NULL,"",SW_SHOWNORMAL);

Application->Terminate();

}

}
//-----

void __fastcall TForm1::Timer1Timer(TObject *Sender)
{

//timer de control de aciertos y desaciertos ventana 2

if((blanco2==1)&& (dispara2==1))//blanco enemigo y disparo efectuado

{

```

```

//instrucciones se ejecutan si condición se cumple
enemigos = enemigos+1;//sumamos enemigos
puntaje=puntaje+300;//sumamos 300 puntos
Edit1->Text=enemigos; //Muestra en Edit1 enemigos caidos
Edit3->Text=puntaje; //Muestra en Edit3 puntaje actual

Image2->Picture->LoadFromFile("sangre1.jpg");
blanco2=2;//no es ni amigo ni enemigo -cambia status de blanco
ALWavePlayer4->Start(); //grito de otro

} //fin del if blanco enemigo y disparo efectuado

if((blanco2==0)&& (dispara2==1)) //blanco amigo y disparo efectuado

{

//instrucciones se ejecutan si condición se cumple
amigos = amigos+1;//sumamos amigos
puntaje=puntaje-900;//restamos 900 puntos
Edit2->Text=amigos; //Muestra en Edit2 amigos caidos
Edit3->Text=puntaje; //Muestra en Edit3 puntaje actual

Image2->Picture->LoadFromFile("sangre1.jpg");
ALWavePlayer6->Start(); //grito de amigo

blanco2=2;//no es ni amigo ni enemigo -cambia status de blanco

} //fin del if blanco amigo y disparo efectuado

if((blanco2==1)&& (dispara2==0)) //pierdo vidas

{

//instrucciones se ejecutan si condición se cumple
// ALWavePlayer6->Start(); // Disparo del enemigo
vidas=vidas-1;//restamos una vida

//control de vidas

switch (vidas) //Borra vidas perdidas
{

case 0:// borra vida 1
    UImage23->Visible=false;
    ALWavePlayer5->Start(); //grito soldado
    Timer10->Enabled=true; //inicia Timer 10 muerte del combatiente

break;

case 1:// borra vida 2
    UImage22->Visible=false;
    ALWavePlayer5->Start(); //grito soldado

break;

case 2:// borra vida 3
    UImage21->Visible=false;
    ALWavePlayer5->Start(); //grito soldado

```



```

break;

case 3:// borra vida4
    ULImage20->Visible=false;
    ALWavePlayer5->Start(); //grito soldado
break;

    } //fin del switch-case de borrar vidas perdidas

    } //fin del if pierdo vidas

if((blanco2==3)&& (dispara2==1)&& ((cargadores<4)|| (granadas<4)) ) //gana armas si me quedan menos de 4
cargadores o menos de 4 granadas

{
    Image2->Picture->LoadFromFile("sangre1.jpg");
    cargadores=cargadores+2;//sumamos 2 cargadores
    granadas=granadas+2;//sumamos 2 granadas

    ULImage7->Visible=true;
    ULImage8->Visible=true;
    ULImage9->Visible=true;
    ULImage10->Visible=true;

    ULImage11->Visible=true;
    ULImage12->Visible=true;
    ULImage13->Visible=true;
    ULImage14->Visible=true;
    ALWavePlayer7->Start();// ruido de mas armas -TADA

    } //fin del if gana armas

dispara2=0;//fin de disparo2 util para rifle y granada
Timer11->Enabled=false;// apago timer de control de aciertos y desaciertos

}
//-----

void __fastcall TForm1::Timer12Timer(TObject *Sender)
{
    //Timer que inicia el cambio de la imagen en puerta 3

    if ( ULImage3->Visible==false)
    {

        switch (numblanco)
        {
            case 1:// imagen de
                Image3->Picture->LoadFromFile("nuevavida.jpg");
                blanco3=3;//da vidas
                break;

            case 2:// imagen de
                Image3->Picture->LoadFromFile("HomeroApunt.jpg");
                blanco3=1;//enemigo
                break;

```

```
case 3:// imagen de
    Image3->Picture->LoadFromFile("zombi1.jpg");
    blanco3=1;//enemigo
break;

case 4:// imagen de

    Image3->Picture->LoadFromFile("bebe1.jpg");
    blanco3=0;//amigo
break;

case 5:// imagen de

    Image3->Picture->LoadFromFile("Gerrillero1.jpg");
    blanco3=1;//enemigo
break;

case 6:// imagen de

    Image3->Picture->LoadFromFile("puerta_abierta.jpg");
    blanco3=2;// ni amigo ni enemigo
break;
case 7:// imagen de

    Image3->Picture->LoadFromFile("BartApunta.jpg");
    blanco3=1;// enemigo
break;

case 8:// imagen de
    Image3->Picture->LoadFromFile("Gerrillero2.jpg");
    blanco3=1;//enemigo
break;

case 9:// imagen de
    Image3->Picture->LoadFromFile("nuevavida.jpg");
    blanco3=3;// da vidas

break;

case 10:// imagen de
    Image3->Picture->LoadFromFile("Soldado1.jpg");
    blanco3=1;//enemigo
break;

    case 11:// imagen de
    Image3->Picture->LoadFromFile("Senoranino4.jpg");
    blanco3=0;// amigo

break;

    case 12:// imagen de
    Image3->Picture->LoadFromFile("zombiArmado2.jpg");
    blanco3=1;// enemigo
break;

    case 13:// imagen de
    Image3->Picture->LoadFromFile("zombiArmado.jpg");
    blanco3=1;//enemigo

break;
```

```

        case 14:// imagen de
        Image3->Picture->LoadFromFile("limpieza2.jpg");
        blanco3=0;// amigo

        break;

    }

    Timer13->Enabled=true; // inicia timer de control de aciertos y desaciertos ventana3

} //fin del if

}
//-----

void __fastcall TForm1::Timer13Timer(TObject *Sender)
{
    //timer de control de aciertos y desaciertos ventana 3

    if((blanco3==1)&& (dispara3==1))//blanco enemigo y disparo efectuado

    {

        //instrucciones se ejecutan si condición se cumple
        enemigos = enemigos+1;//sumamos enemigos
        puntaje=puntaje+300;//sumamos 300 puntos
        Edit1->Text=enemigos; //Muestra en Edit1 enemigos caidos
        Edit3->Text=puntaje; //Muestra en Edit3 puntaje actual

        Image3->Picture->LoadFromFile("sangre1.jpg");
        blanco3=2;//no es ni amigo ni enemigo -cambia status de blanco
        ALWavePlayer4->Start(); //grito de otro

    } //fin del if blanco enemigo y disparo efectuado

    if((blanco3==0)&& (dispara3==1)) //blanco amigo y disparo efectuado

    {

        //instrucciones se ejecutan si condición se cumple
        amigos = amigos+1;//sumamos amigos
        puntaje=puntaje-900;//restamos 900 puntos
        Edit2->Text=amigos; //Muestra en Edit2 amigos caidos
        Edit3->Text=puntaje; //Muestra en Edit3 puntaje actual

        Image3->Picture->LoadFromFile("sangre1.jpg");
        ALWavePlayer6->Start(); //grito de amigo

        blanco3=2;//no es ni amigo ni enemigo -cambia status de blanco

    } //fin del if blanco amigo y disparo efectuado

    if((blanco3==1)&& (dispara3==0)) //pierdo vidas

    {

        //instrucciones se ejecutan si condición se cumple
        // ALWavePlayer6->Start(); // Disparo del enemigo
        vidas=vidas-1;//restamos una vida
    }
}

```

```

//control de vidas

switch (vidas) //Borra vidas perdidas
{

case 0:// borra vida 1
    ULImage23->Visible=false;
    ALWavePlayer5->Start(); //grito soldado
    Timer10->Enabled=true; //inicia Timer 10 muerte del combatiente

    break;

case 1:// borra vida 2
    ULImage22->Visible=false;
    ALWavePlayer5->Start(); //grito soldado

    break;
case 2:// borra vida 3
    ULImage21->Visible=false;
    ALWavePlayer5->Start(); //grito soldado
    break;

case 3:// borra vida4
    ULImage20->Visible=false;
    ALWavePlayer5->Start(); //grito soldado
    break;

} //fin del switch-case de borrar vidas perdidas

} //fin del if pierdo vidas

if((blanco3==3)&& (dispara3==1)&& (vidas<4) ) //gana vidas cuando me quedan menos de 4 vidas
{
    Image3->Picture->LoadFromFile("sangre1.jpg");
    vidas=vidas+2;//sumamos dos vida
    ULImage20->Visible=true;
    ULImage21->Visible=true;
    ULImage22->Visible=true;
    ULImage23->Visible=true;
    ALWavePlayer7->Start();// ruido de vida -TADA

} //fin del if gana vidas

dispara3=0;// fin de disparo3 util para rifle y granada

Timer13->Enabled=false;// apago timer de control de aciertos y desaciertos vrntana 3

}
//-----

void __fastcall TForm1::Timer14Timer(TObject *Sender)
{
//Analisis de rango segun puntaje

if(puntaje<2000)

{

```

```

//instrucciones se ejecutan si condición se cumple
Edit4->Text="Soldado";
}

if((puntaje<4000)&&(puntaje>2000))

{

//instrucciones se ejecutan si condición se cumple
Edit4->Text="Cabo";
}

if((puntaje>4000)&&(puntaje<9000))

{

//instrucciones se ejecutan si condición se cumple
Edit4->Text="Sargento";
}

if((puntaje>9000)&&(puntaje<20000))

{

//instrucciones se ejecutan si condición se cumple
Edit4->Text="Coronel";
}

if((puntaje>20000)&&(puntaje<30000))

{

//instrucciones se ejecutan si condición se cumple
Edit4->Text="General";
}

if(puntaje>30000)

{

//instrucciones se ejecutan si condición se cumple
Edit4->Text="General Retirado";
Timer10->Enabled=true; //Timer de muerte o victoria
}

}
//-----

void __fastcall TForm1::Timer15Timer(TObject *Sender)
{

ALWavePlayer8->Enabled=true; // ruido de tome rifle

Timer15->Enabled=false; //grito del rifle
}
//-----

```

FIN DEL LISTADO