Ministerul Educaţiei al Republicii Moldova

Universitatea Tehnică a Moldovei

Catedra: Automatica și Tehnologii Informaționale

**RAPORT**

Lucrare de laborator Nr.1

*la M.I.D.P.S.*

A efectuat: st. gr. TI-144

I. Stratan

A verificat: lector.univ.

S. Cojocaru

Chişinău 2016

**Lucrarea de laborator 1**

**MEDIUL INTEGRAT C++ BUILDER**

**Obiectivele lucrării**

**a)** Însuşirea modului de utilizare a celor mai importante componente ale mediului integrat C++ BUILDER . Realizarea unui program simplu care utilizează componente de tip *TButton, TEdit, Tlabel, RadioButton*  etc.

**b)** Însuşirea modului de utilizare a componentei VCL **TTimer.** Însuşirea modului de utilizare a funcţiilor de lucru cu timpul sistem. Realizarea unor aplicaţii de gestionare a resursei timp.

**c)** Însuşirea modului de utilizare a componentelor VCL **TPaintBox** şi **TPanel.** Însuşirea modului de utilizare a principalelor funcţii grafice ale mediului C++BUILDER . Realizarea unor elemente pentru afişarea grafică a informaţiei (diagramă şi bargraf).

**Facilităţile mediului C++Builder**

Borland C++Builder este un mediu de programare vizual, orientat obiect, pentru dezvoltarea rapidă de aplicaţii (**RAD**) cu scop general şi aplicaţii client/server pentru Windows şi WindowsNT. Folosind C++Builder se pot crea aplicaţii Windows eficiente sciind un minim de cod. Facilităţile semnificative oferite de acestea sunt prezentate succint în cele ce urmează.

# Înalta productivitate a mediului de dezvoltare

Aceasta este favorizată de principalele instrumente furnizate de mediul de dezvoltare integrat (**IDE**) C++Builder şi anume :

*- Visual Form Designer;*

*- Object Inspector;*

*- Component Palette;*

*- Project Manager;*

*- Code Editor;*

*- Debugger.*

Acestea dau posibilitatea utilizatorului să dezvolte rapid aplicaţii având totodată un control complet asupra codului şi resurselor.

1. **IDE (Mediul de Dezvoltare Integrat)**

Elementele mediului integrat de dezvoltare sunt:

* Meniu principal (Main Menu);
* Forma (Form);
* Editorul de cod (Code Editor);
* Bara cu instrumente (Toolbar);
* Paleta cu componente (Component Palette);
* Tabelul cu proprietăţi ale obiectelor (Object Inspector);
* Administratorul de program (Program Manager).

**A)**

int i=0;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

Edit1->Text=0;

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button3Click(TObject \*Sender)

{

Close();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

i=i+1;

Edit1->Text=Edit1->Text.sprintf("Incrementare %d",i);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button2Click(TObject \*Sender)

{

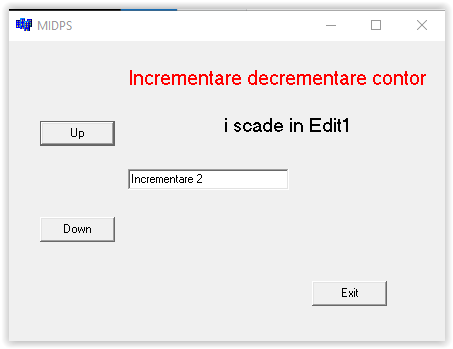
i=i-1;

Edit1->Text=Edit1->Text.sprintf("Decrementare %d",i);

}

//---------------------------------------------------------------------------

**Verificare :**



**B)**

TForm1 \*Form1;

int i=0;

struct date d;

struct time t;

struct time a;

struct Timp {

int m;

int s;

int z;

} ;

Timp myTime;

void PrintTime();

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

Timer1->Enabled=false;

Button1->Enabled=true;

Button2->Enabled=true;

Button3->Enabled=false;

PrintTime();

}

//---------------------------------------------------------------------------

void PrintTime()

{

stringstream timeFormat;

timeFormat << ((myTime.m < 10) ? "0" : "") << myTime.m << " min ";

timeFormat << ((myTime.s < 10) ? "0" : "") << myTime.s << " sec ";

timeFormat << myTime.z << " zec";

string temp = timeFormat.str();

Form1->Edit1->Text = (AnsiString)temp.c\_str();

}

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

Timer1->Enabled=true;

Button1->Enabled=false;

Button2->Enabled=true;

Button3->Enabled=false;

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button2Click(TObject \*Sender)

{

Timer1->Enabled=false;

Button1->Enabled=true;

Button2->Enabled=false;

Button3->Enabled=true;

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Timer1Timer(TObject \*Sender)

{

if(myTime.z == 9)

{

myTime.z = 0;

if(myTime.s == 59)

{

myTime.s = 0;

myTime.m++;

}

else myTime.s++;

}

else myTime.z++;

PrintTime();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button4Click(TObject \*Sender)

{

Close();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button3Click(TObject \*Sender)

{

Edit1->Clear();

myTime.m=0;

myTime.s=0;

myTime.z=0;

PrintTime();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Timer2Timer(TObject \*Sender)

{

char buf[20];

getdate(&d);

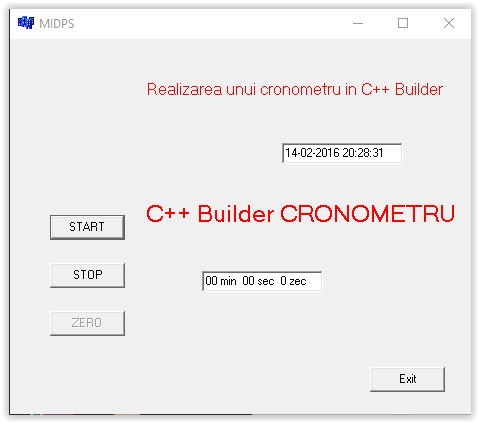
gettime(&t);

sprintf(buf,"%02d-%02d-%4d %02d:%02d:%02d",d.da\_day,d.da\_mon,d.da\_year,t.ti\_hour,t.ti\_min,t.ti\_sec);

Edit2->Text=(AnsiString)buf;

}

**Verificare :**



**C)**

TForm1 \*Form1;

struct date d;

struct time t;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Timer1Timer(TObject \*Sender)

{

char buf[20];

getdate(&d);

gettime(&t);

sprintf(buf,"%02d-%02d-%4d %02d:%02d:%02d",d.da\_day,d.da\_mon,d.da\_year,

t.ti\_hour,t.ti\_min,t.ti\_sec);

Edit1->Text=(AnsiString)buf;

}

void \_\_fastcall TForm1::Button3Click(TObject \*Sender)

{

Close();

}

void \_\_fastcall TForm1::PaintBox1Click(TObject \*Sender)

{

// PaintBox1->Canvas->Pen->Color = clBlack;

PaintBox1->Canvas->Brush->Color = clBlack;

PaintBox1->Canvas->Brush->Style = bsCross;

PaintBox1->Canvas->Rectangle(0,250,250,0);

}

//---------------------------------------------------------------------------

int width;

int height;

int x;

int y;

void DrawLine()

{

y = (height / 2.0) + (rand() % 41 - 20);

Form1->PaintBox1->Canvas->LineTo(++x, y);

Form1->Panel2->Height = y;

}

void DrawLine(int width, int height);

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

PaintBox1->Repaint();

PaintBox1->Canvas->Pen->Color = clRed;

width = Form1->PaintBox1->Width;

height = Form1->PaintBox1->Height;

x = 0;

Form1->PaintBox1->Canvas->MoveTo(0, height / 2.0);

Button2->Enabled = true;

Button1->Enabled = false;

Timer2->Enabled = true;

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Timer2Timer(TObject \*Sender)

{

if(x == width)

Button2Click(Sender);

DrawLine();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button2Click(TObject \*Sender)

{

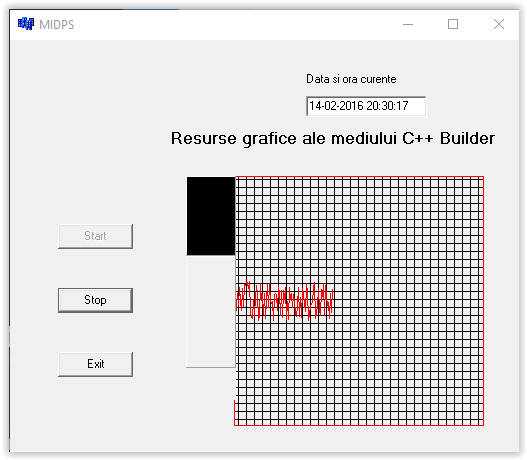
Timer2->Enabled = false;

Button1->Enabled = true;

Button2->Enabled = false;

}

**Verificare :**



**Concluzie**

În urma efectuării acestui laborator , am facut cunoștiință mediul de dezvoltare C++ Builder, am înțeles cum funcționează componentele TButton, Ttimer, Label, Edit, PaintBox, Panel etc. Am văzut cum se creează un proiect. Și a fost interesant să lucrez cu aceste obiecte. C++ Builder este destul de perfomant și conține tot de ce e nevoie pentru a crea p fereastră ce stă la baza unui program.