

FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

UNIVERSITATEA TEHNICĂ A MOLDOVEI

MEDII INTERACTIVE DE DEZVOLTARE A PRODUSELOR SOFT

LUCRARE DE LABORATOR №1

MEDIUL INTEGRAT C++ BUILDER

Autor:

st. gr. TI-141

BULDUMAC VASILE

lector asistent:

Irina COJANU

lector superior:

Svetlana COJOCARU

LUCRARE DE LABORATOR #1

1. Scopul lucrării

Studierea modului de utilizare a celor mai importante componente ale mediului integrat C++ BUILDER.

2. Obiectivele lucrării

- a) Însușirea 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) Însușirea modului de utilizare a componentei VCL TTimer. Însușirea modului de utilizare a funcțiilor de lucru cu timpul sistem. Realizarea unor aplicații de gestionare a resursei timp.
- c) Însușirea modului de utilizare a componentelor VCL TPaintBox și TPanel. Însușirea modului de utilizare a principalelor funcții grafice ale mediului C++BUILDER. Realizarea unor elemente pentru afișarea grafică a informației (diagramă și bargraf).

3. Efectuarea lucrării de laborator

3.1. Task-uri implementate

1. Se elaborează un program pentru realizarea unui contor cu funcțiile incrementare/decrementare.
2. Se elaborează un program pentru realizarea unui cronometru.
3. Se elaborează un program pentru realizarea a două elemente de afișare (bargraf și diagramă cu avans continuu).

3.2. Analiza lucrării de laborator

<https://github.com/mowshon/MIDPS.git>

- a) Realizarea unui program simplu care utilizează componente de tip *TButton*, *TEdit*, *Tlabel*, *RadioButton* etc. (Fig. a1)

```
#include <vcl.h>
#pragma hdrstop
```

```
#include "Unit1.h"
```

```
#pragma package(smart_init)
```

```
#pragma resource "*.dfm"
```

```
TForm1 *Form1;
```

```
int j = 0;
```

```
__fastcall TForm1::TForm1(TComponent* Owner)
```

```
    : TForm(Owner)
```

```
{
```

```
}
```

```
void __fastcall TForm1::Button1Click(TObject *Sender)
```

```
{
```

```
    j = j + 1;
```

```
    Label1->Caption = "J se incrementeaza";
```

```
    Edit1->Text = j;
```

```
}
```

```
void __fastcall TForm1::Button2Click(TObject *Sender)
```

```
{
```

```
    j = j - 1;
```

```
    Label1->Caption = "J se decrementeaza";
```

```
    Edit1->Text = j;
```

```
}
```

```
void __fastcall TForm1::Button3Click(TObject *Sender)
```

```
{
```

```
    Close();
```

```
}
```

b) Realizarea unor aplicații de gestionare a resursei timp. (Fig. b1)

```
#include <vcl.h>
#pragma hdrstop

#include "Unit1.h"
#include "stdio.h"
#include "dos.h"

#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;

struct date d;
struct time t;

int zecimi = 0;
int secunde = 0;
int minute = 0;

__fastcall TForm1::TForm1(TComponent* Owner)
    : TForm(Owner)
{
    Label1->Caption = "Realizarea unui cronometru in Builder C++";
    Label2->Caption = "Cronometru C++ Builder";
    Timer1->Enabled = false;
    Edit1->Clear();
    Edit2->Clear();
}
```

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

```
{  
    Close();  
}
```

```
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;  
  
}
```

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

```
{  
    zecimi++;  
    if(zecimi > 9)  
    {  
        secunde++;  
        zecimi = 0;  
    }  
    if(secunde > 59)  
    {  
        minute++;  
        secunde = 0;  
    }  
    char buf[20];  
    sprintf(buf,"%02d:%02d:%02d",minute, secunde, zecimi);
```

```

        Edit1->Text = (AnsiString)buf;
    }

    void __fastcall TForm1::StartClick(TObject *Sender)
    {
        Timer1->Enabled = true;
    }

    void __fastcall TForm1::StopClick(TObject *Sender)
    {
        Timer1->Enabled = false;
    }

    void __fastcall TForm1::ZeroClick(TObject *Sender)
    {
        zecimi = 0;
        secunde = 0;
        minute = 0;
        Edit1->Text = "00:00:00";
    }

```

c) Realizarea unor elemente pentru afișarea grafică a informației (diagramă și bargraf). (Fig. c1)

```

#include <vcl.h>
#pragma hdrstop

#include "Unit1.h"
#include "stdio.h"
#include "dos.h"

```

```
#include "stdlib.h"
```

```
#pragma package(smart_init)
```

```
#pragma resource "*.dfm"
```

```
TForm1 *Form1;
```

```
struct date d;
```

```
struct time t;
```

```
int cordx = 0;
```

```
__fastcall TForm1::TForm1(TComponent* Owner)
```

```
    : TForm(Owner)
```

```
{
```

```
    Edit1->Clear();
```

```
    Timer2->Enabled = false;
```

```
}
```

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

```
{
```

```
    Close();
```

```
}
```

```
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::StartClick(TObject *Sender)
{
    Timer2->Enabled = true;
}

```

```

void __fastcall TForm1::StopClick(TObject *Sender)
{
    Timer2->Enabled = false;
}

```

```

void __fastcall TForm1::Timer2Timer(TObject *Sender)
{
    PaintBox1->Canvas->Brush->Color = clGray;
    PaintBox1->Canvas->Pen->Color = clGray;
    PaintBox1->Canvas->Brush->Style = bsCross;
    PaintBox1->Canvas->Rectangle(0,0,PaintBox1->Width,PaintBox1->Height);
    PaintBox1->Canvas->FloodFill(PaintBox1->Left+5,PaintBox1->Top+5,clBlack,fsBorder);

    Panel2->Height = rand() % 150 + 50;
    PaintBox1->Canvas->Pen->Color=clLime;
    PaintBox1->Canvas->Pen->Width = 1;
}

```



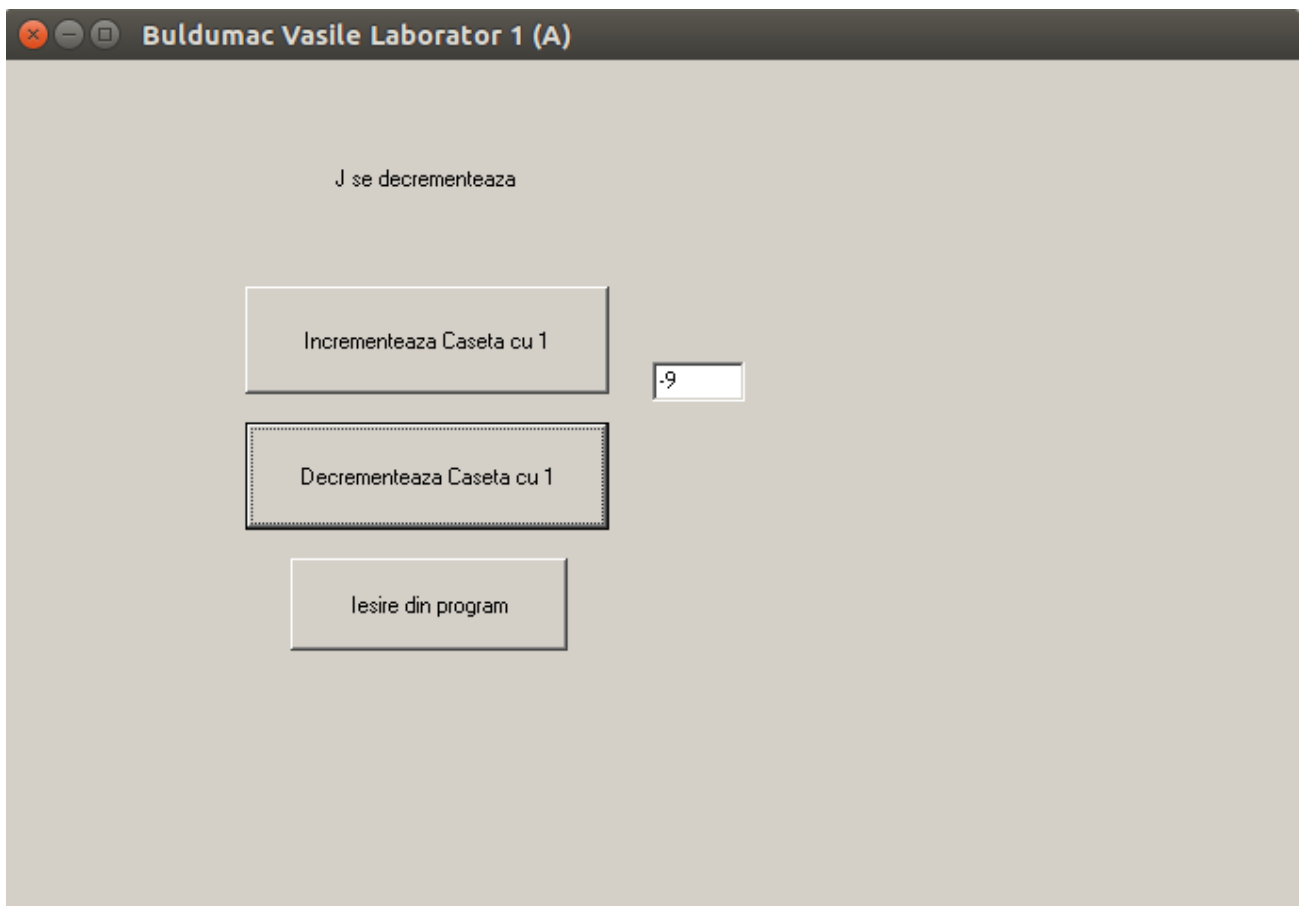
```

PictureBox1->Canvas->LineTo(cordx,rand() % 150 + 50);
cordx += 10;
if(cordx > PictureBox1->Width){
    cordx = 0;
    PictureBox1->Canvas->MoveTo(0,150);
    PictureBox1->Repaint();
}
}

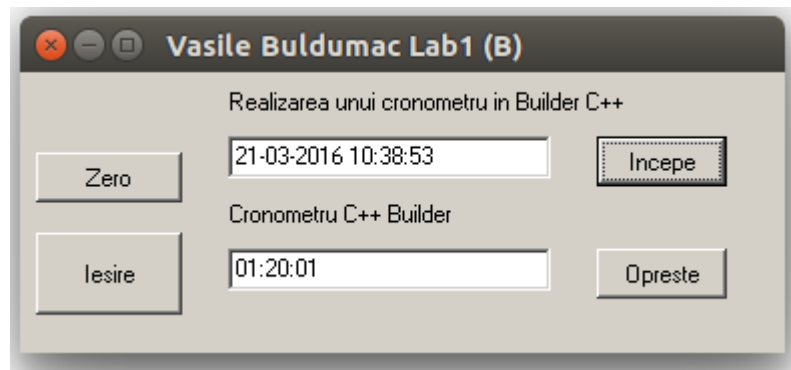
```

3.3. Rezultatele finale primite

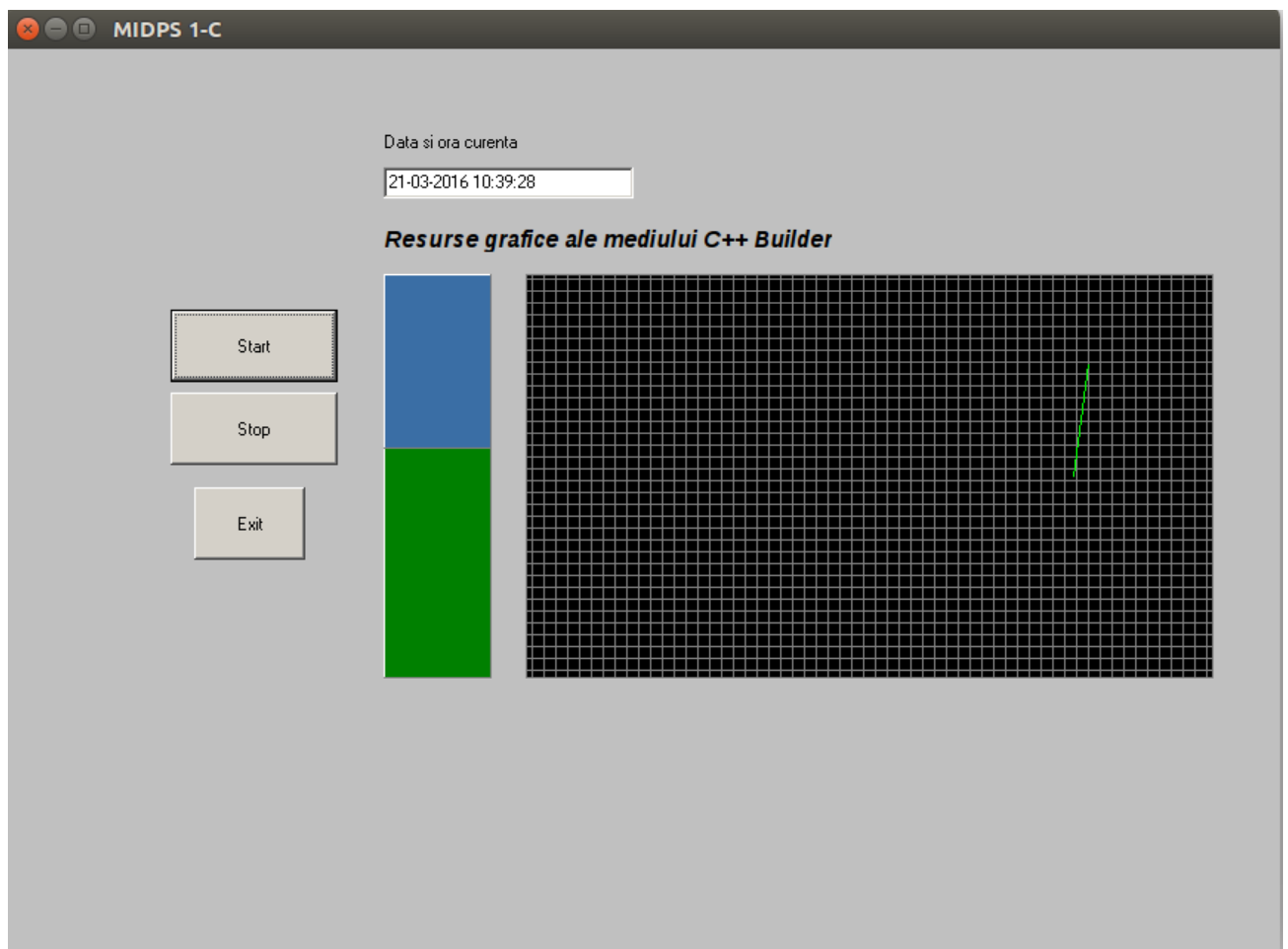
1. Fig. A1



2. Fig. b1



3. Fig. c1



Concluzie

În urma realizării laboratorului nr.1 la tema: ”*Mediul Integrat C++ Builder*”, am însușit modul de utilizare a celor mai importante componente ale Mediului Integrat C++ BUILDER.

La punctul (a), am realizat un program simplu care utilizează componente de tip *TButton*, *TEdit*, *Tlabel*, *RadioButton*.

- *Incrementeaza Caseta cu 1* – incrementarea variabilei *j*;
- *Decrementeaza Caseta cu 1* - decrementarea variabilei *j*;
- *Ieșire din program* – oprește programul și închide fereastra.

Fereastra mai include o casetă de editare unde se afișează valoarea variabile *j*

La punctul (b), am elaborat un program care utilizează componenta de tip *VCL TTimer*, pentru realizarea unui cronometru.

Programul conține 4 butoane, astfel realizând funcția dată în condiție:

- *Incepe* – pornirea cronometrului;
- *Opreste* - oprirea cronometrului;
- *Zero* – inițializarea cronometrului;
- *Exit* - oprește rularea programului și închide fereastra.

Fereastra include două timer-e, pentru afișarea timpului curent și pentru cronometru, două etichete, care afișează textele: ”*Realizarea unui cronometru în BuilderC++*”, ”*Cronometru C++ Builder*”, corespunzătoare celor două casete de editare de mai jos.

La punctul (c), am elaborat un program care utilizează componentele de tip *VCL TPaintBox* și *TPanel*, pentru realizarea a două elemente de afișare (bargraf și diagramă cu avans continuu).

- *Start* – activarea afișării în diagramă și în bargraf;
- *Stop* - oprirea afișării în diagramă și în bargraf;
- *Exit* - oprește rularea programului și închide fereastra.

În concluzie, am studiat modul de utilizare a componentelor de bază, a funcțiilor de lucru cu timpul sistem, cât și utilizarea funcțiilor grafice la realizarea unor elemente pentru afișarea grafică a informației.

Bibliografie

1. Îndrumar metodologic pentru lucrările de laborator la obiectul MIDPS
2. <http://www.functionx.com/bcb/>
3. <http://www.cprogramming.com/borland.html>
4. <http://www.tenouk.com/cncpluspluscompiler.html>
5. http://www.ee.ic.ac.uk/pcheung/teaching/ee2_software_engineering/Lesson1%20-%20Introduction%20to%20BCB.pdf

