Технічне Забезпечення Інформаційних Технологій

Лабораторна робота №3

«Дослідження штрихового коду EAN 13»

Виконав:

Студент групи

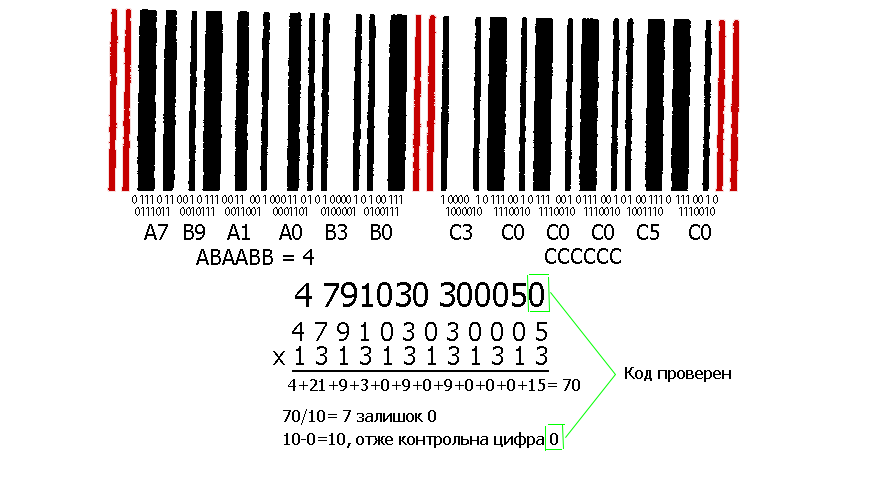
ДА-21

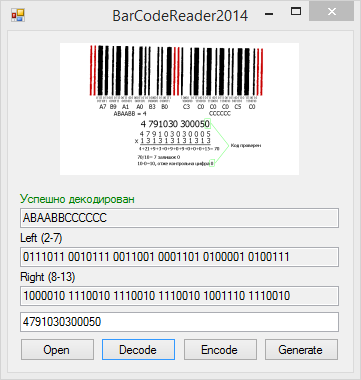
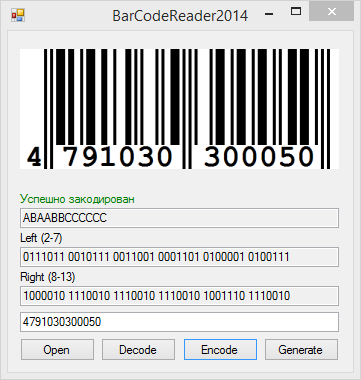
Воронін Ігор

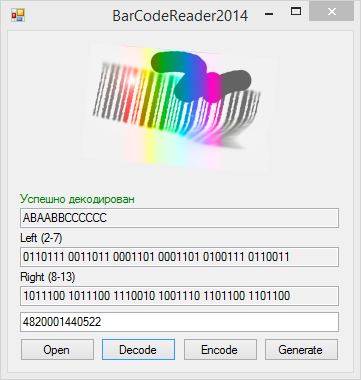
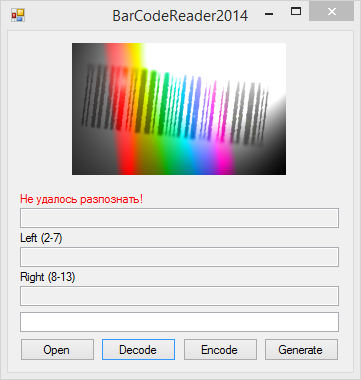
Київ – 2014

*Мета: дослідження основних принципів побудови штрихового коду EAN*

1. Результати ручного декодування штрихового коду.



1. Результати автоматичного декодування й кодування штрихового коду.  
2. Складні й дуже складні випадки

1. Лістинг коду програми

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace BarCodeReader2014

{

public partial class Form1 : Form

{

Bitmap Img;

Dictionary<string, number\_type> encoding\_table = new Dictionary<string, number\_type>();

string[] A\_codes = new string[10] { "0001101", "0011001", "0010011", "0111101", "0100011", "0110001",

"0101111","0111011","0110111","0001011"};

string[] first\_number\_code = new string[10]{"AAAAAA", "AABABB", "AABBAB", "AABBBA", "ABAABB", "ABBAAB",

"ABBBAA", "ABABAB", "ABABBA", "ABBABA"};

public Form1()

{

InitializeComponent();

for (int i = 0; i < 10; i++)

{

encoding\_table.Add(str\_mutate(A\_codes[i], 'A'), new number\_type(i, 'A'));

encoding\_table.Add(str\_mutate(A\_codes[i], 'B'), new number\_type(i, 'B'));

encoding\_table.Add(str\_mutate(A\_codes[i], 'C'), new number\_type(i, 'C'));

}

}

private void OpenBtn\_Click(object sender, EventArgs e)

{

DialogResult Res = OpenDlg.ShowDialog();

if (!Res.HasFlag(DialogResult.Cancel) && System.IO.File.Exists(OpenDlg.FileName))

{

Img = new Bitmap(OpenDlg.FileName);

PicBox.Image = Img;

}

}

private bool IsBlack(Color c)

{

return (c.R \* 0.3 < 120 && c.B \* 0.11 < 120 && c.G \* 0.59 < 120);

}

struct number\_type

{

public number\_type(int n, char t)

{

number = n;

type = t;

}

public int number;

public char type;

}

private bool TryDecode(string full\_bin\_list)

{

BinaryBox1.Clear();

BinaryBox2.Clear();

PatterBox.Clear();

number\_type[] n\_type\_array = new number\_type[12];

for (int i = 0; i < 12; i++)

{

if (!encoding\_table.TryGetValue(full\_bin\_list.Substring(i \* 7, 7), out n\_type\_array[i]))

{

label1.Text = "Error. There is no such bit patter in encoding table";

return false;

}

if (i < 6)

BinaryBox1.Text += full\_bin\_list.Substring(i \* 7, 7) + " ";

else

BinaryBox2.Text += full\_bin\_list.Substring(i \* 7, 7) + " ";

}

string encoded\_type\_str = "";

for (int i = 0; i < 6; i++)

{

encoded\_type\_str += n\_type\_array[i].type.ToString();

}

for (int i = 0; i < 12; i++)

{

PatterBox.Text += n\_type\_array[i].type.ToString();

}

int first\_number = -1;

for (int i = 0; i < 10; i++)

{

if (first\_number\_code[i] == encoded\_type\_str)

{

first\_number = i;

break;

}

}

if (first\_number == -1)

{

label1.Text = "Error. First number wasn't recognized";

return false;

}

string fullEAN = first\_number.ToString();

for (int i = 0; i < 12; i++)

{

fullEAN += n\_type\_array[i].number.ToString();

}

DigitBox.Text = fullEAN;

if (get\_control\_sum(fullEAN) != int.Parse(fullEAN[12].ToString()))

{

label1.Text = "Error. Checksum is wrong";

return false;

}

return true;

}

private void DrawLine(int Pos, int barwidth, Bitmap b)

{

for (int y = 0; y < b.Height; y++)

for (int x = barwidth \* Pos; x < barwidth \* Pos + barwidth; x++)

b.SetPixel(x, y, Color.Black);

}

private void Encode(string ean)

{

Bitmap Img2 = new Bitmap(530, 200);

using (Graphics g = Graphics.FromImage(Img2))

{

int barwidth = 5;

int shift = 7;

g.Clear(Color.White);

DrawLine(1 + shift, barwidth, Img2);

DrawLine(3 + shift, barwidth, Img2);

DrawLine(47 + shift, barwidth, Img2);

DrawLine(49 + shift, barwidth, Img2);

DrawLine(95 + shift, barwidth, Img2);

DrawLine(93 + shift, barwidth, Img2);

PatterBox.Text = first\_number\_code[int.Parse(ean[0].ToString())] + "CCCCCC";

ean += get\_control\_sum(ean).ToString();

DigitBox.Text = ean;

string bin\_code = "";

for (int i = 1; i < 7; i++)

{

bin\_code += str\_mutate(A\_codes[int.Parse(ean[i].ToString())], first\_number\_code[int.Parse(ean[0].ToString())][i - 1]);

}

for (int i = 7; i < 13; i++)

{

bin\_code += str\_mutate(A\_codes[int.Parse(ean[i].ToString())], 'C');

}

BinaryBox1.Clear();

BinaryBox2.Clear();

for (int i = 0; i < 6; i++)

{

BinaryBox1.Text += bin\_code.Substring(i \* 7, 7) + " ";

}

for (int i = 6; i < 12; i++)

{

BinaryBox2.Text += bin\_code.Substring(i \* 7, 7) + " ";

}

for (int i = 0; i < 42; i++)

{

if (bin\_code[i] == '1') DrawLine(4 + i + shift, barwidth, Img2);

}

for (int i = 42; i < 84; i++)

{

if (bin\_code[i] == '1') DrawLine(9 + i + shift, barwidth, Img2);

}

g.FillRectangle(Brushes.White, (4 + shift) \* barwidth, Img2.Height - 40, 42 \* barwidth, 40);

g.FillRectangle(Brushes.White, (51 + shift) \* barwidth, Img2.Height - 40, 42 \* barwidth, 40);

g.DrawString(ean.Substring(1, 6), new Font("Courier New", 38, FontStyle.Bold), Brushes.Black, (5 + shift) \* barwidth, Img2.Height - 47);

g.DrawString(ean.Substring(7, 6), new Font("Courier New", 38, FontStyle.Bold), Brushes.Black, (52 + shift) \* barwidth, Img2.Height - 47);

g.DrawString(ean[0].ToString(), new Font("Courier New", 38, FontStyle.Bold), Brushes.Black, (-7 + shift) \* barwidth, Img2.Height - 47);

}

Img = Img2;

PicBox.Image = Img2;

}

private void DecodeBtn\_Click(object sender, EventArgs e)

{

if (Img == null) return;

List<List<int>> bb = new List<List<int>>();

for (int yi = 0; yi < Img.Height; yi++)

{

List<int> a = new List<int>();

int b = 1;

bool bc = IsBlack(Img.GetPixel(0, yi));

for (int xi = 1; xi < Img.Width; xi++)

{

bool bc2 = IsBlack(Img.GetPixel(xi, yi));

if (bc2 != bc)

{

a.Add(b);

b = 0;

}

bc = bc2;

b++;

}

if (a.Count == 60)

bb.Add(a);

}

if (bb.Count == 0 )

{

label1.Text = "Не удалось разпознать!";

label1.ForeColor = Color.Red;

return;

}

double b\_len = 0;

foreach (List<int> a in bb)

{

b\_len += a[1] + a[3] + a[a.Count - 1] + a[a.Count - 3];

}

b\_len /= 4 \* bb.Count;

double w\_len = 0;

foreach (List<int> a in bb)

{

w\_len += a[2] + a[a.Count - 3];

}

w\_len /= 2 \* bb.Count;

int bb\_inner\_length = bb[0].Count;

string full\_bin\_list = "";

for (int i = 4; i <= bb\_inner\_length - 4; i++)

{

int cur\_sum = 0;

foreach (List<int> cur\_line in bb)

{

cur\_sum += cur\_line[i];

}

int int\_bits\_amount = 0;

double average\_length = (double)cur\_sum / bb.Count;

if (i % 2 == 0)

{

double bit\_amount = (double)average\_length / w\_len;

int\_bits\_amount = (int)Math.Round(bit\_amount);

for (int j = 0; j < int\_bits\_amount; j++)

{

full\_bin\_list += "0";

}

}

else

{

double bit\_amount = (double)average\_length / b\_len;

int\_bits\_amount = (int)Math.Round(bit\_amount);

for (int j = 0; j < int\_bits\_amount; j++)

{

full\_bin\_list += "1";

}

}

}

if (full\_bin\_list.Length < 89)

{

label1.Text = "Не удалось разпознать!";

label1.ForeColor = Color.Red;

return;

}

full\_bin\_list = full\_bin\_list.Substring(0, 42) + full\_bin\_list.Substring(47, 42); // Видалємо середні допоміжні біти

label1.Text = "Успешно декодирован";

label1.ForeColor = Color.Green;

if (!TryDecode(full\_bin\_list))

{

string s = "";

foreach (char c in full\_bin\_list) s = c.ToString() + s;

full\_bin\_list = s;

label1.Text = "Успешно декодирован";

if (!TryDecode(full\_bin\_list))

{

label1.ForeColor = Color.Red;

}

}

}

private int get\_control\_sum(string source\_digits)

{

int sum\_evens = 0;

int sum\_odds = 0;

for (int i = 0; i < 12; i++)

{

if ((i % 2) == 1)

{

sum\_evens += int.Parse(source\_digits[i].ToString());

}

else

{

sum\_odds += int.Parse(source\_digits[i].ToString());

}

}

return (10 - ((sum\_odds + 3 \* sum\_evens) % 10)) % 10;

}

private string str\_mutate(string source, char param)

{

if (param == 'A')

return source;

string res = "";

for (int i = 0; i < source.Length; i++)

{

char c;

if (source[i] == '0')

{

c = '1';

}

else

{

c = '0';

}

if (param == 'B')

res = c.ToString() + res;

else if (param == 'C')

res = res + c.ToString();

}

return res;

}

private void button1\_Click(object sender, EventArgs e)

{

if (DigitBox.Text.Length > 11)

{

Encode(DigitBox.Text.Substring(0, 12));

label1.ForeColor = Color.Green;

label1.Text = "Успешно закодирован";

}

}

private void button2\_Click(object sender, EventArgs e)

{

Random r = new Random();

DigitBox.Clear();

for (int i = 0; i < 12; i++)

DigitBox.Text += r.Next(9).ToString();

DigitBox.Text += get\_control\_sum(DigitBox.Text).ToString();

button1\_Click(sender, e);

}

}

}

***Висновки:***

Під час лабораторної роботи було ознайомлено з структурої штрих кодів EAN-13, методами їх кодування та декодування. Була розроблена программа здатна коректно зчитувати код з картинки, перевіряти його контрольну сумму, а також кодувати довільні коди у штрих код.