Michał Muzyk 218332 Data: 10.12.2016 r.

Jakub Jaskóła 218349

Grupa: Wtorek TN 13:15-16:15

Prowadzący: Dr inż. Jan Nikodem

**Urządzenia peryferyjne**

**Drukowanie kodów paskowych**

**1. Zadania do wykonania**

Celem ćwiczenia było napisanie aplikacji graficznej, która umożliwi generowanie kodów kreskowych typu EAN-13 oraz ich wydrukowania za pomocą drukarki.  
  
**2. Wstęp teoretyczny**

Kod kreskowy to graficzna reprezentacja informacji poprzez kombinację ciemnych i jasnych elementów, ustaloną według symboliki reguł opisujących budowę kodu, (np. jego wymiary, zbiór kodowanych znaków, algorytm obliczania cyfry kontrolnej i inne) danego kodu. Kod kreskowy przeznaczony jest dla czytników elektronicznych. Ma na celu umożliwienie automatycznego odczytywania informacji. Głównym zastosowaniem kodu kreskowego jest automatyczna identyfikacja produktów w szeroko pojętej logistyce.   
 Kod liczbowy EAN-13 składa się z czterech obszarów: 1) system numerowania, 2) kod wytwórcy, 3) kod produktu, 4) cyfra kontrolna. Pierwsza cyfra systemu numerowego jest drukowana na lewo od kodu kreskowego, a druga cyfra systemu jest drukowana jako pierwsza cyfra 6-cio cyfrowej grupy poniżej lewej części kodu kreskowego, kod wytwórcy daje następne cyfry tej grupy. Kod produktu stanowi pierwsze 5 cyfr 'prawej' grupy poniżej kodu kreskowego, a cyfra kontrolna stanowi 6-tą cyfrę tej grupy. By kod EAN-13 mógł zostać automatycznie "przeczytany" przez urządzenia skanujące musi zostać przetworzony do postaci kodu kreskowego. Sposób kodowania jest na pierwszy rzut oka nieco skomplikowany a to wszystko za sprawą kodu UPC-A, z którym musiał być on zgodny. Jest on 12-cyfrowy a EAN-13 jest 13-cyfrowy. Trzeba było zatem zakodować w jakiś sprytny sposób 1 cyfrę, ale o tym za chwilę. Kod kreskowy EAN-13 składa się z 95 białych i czarny prążków znajdujących się jeden przy drugim. Przyjmujemy, że 1 oznacza pasek ciemny, a 0 oznacza pasek jasny.   
 Cyfrę kontrolną (trzynastą) w EAN 13 oblicza się mnożąc pierwszą cyfrę przez 1, drugą przez 3, trzecią przez 1, czwartą przez 3, piątą przez 1, szóstą przez 3, siódmą przez 1, ósmą przez 3, dziewiątą przez 1, dziesiątą przez 3, jedenastą przez 1, dwunastą przez 3. Następnie sumuje się wszystkie iloczyny i sumę dzieli się przez 10, a więc pozostawia się ostatnią cyfrę wyniku. Jeśli brak reszty z tego dzielenia to jest to nasza cyfra kontrolna. W przeciwnym wypadku szuka się dopełnienia do następnej liczby podzielnej przez 10, czyli dzielimy sumę iloczynów modulo 10 i wynik odejmujemy od 10 - jest to nasza cyfra kontrolna.  
  
W projekcie wykorzystywaliśmy funkcje znajdujące się w bibliotece:

\* Zen.Barcode.Core – biblioteka generująca kody EAN-13   
  
oraz klasach takich jak: Bitmap, PrintDocument.  
  
**3. Opisany kod programu**Opiskodu:  
  
private void btnBarcode\_Click(object sender, EventArgs e)

{

Zen.Barcode.CodeEan13BarcodeDraw barcode = Zen.Barcode.BarcodeDrawFactory.CodeEan13WithChecksum; **// ustawienie trybu generowanie kodow EAN-13**

pictureBox1.Image = barcode.Draw(txtBarcode.Text, 100); **// stworzenie obrazu w oknie na podstawie cyfr wprowadzonych w polu tekstowym }**

private void pictureBox1\_Click(object sender, EventArgs e)

{

}

private void myPrintDocument2\_PrintPage(System.Object sender, System.Drawing.Printing.PrintPageEventArgs e)

{

Bitmap myBitmap1 = new Bitmap(pictureBox1.Width, pictureBox1.Height); **// stworzenie bitmapy poprzez pobranie szerokosci i wysokosci okna**

pictureBox1.DrawToBitmap(myBitmap1, new Rectangle(0, 0, pictureBox1.Width, pictureBox1.Height)); **// renderowanie zawartosci okna do mapy bitowej**

e.Graphics.DrawImage(myBitmap1, 0, 0); // rysowanie

myBitmap1.Dispose(); **// zwolnienie zasobow uzywanych przez mape bitowa**

}

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

{

**// stworzenie obiektu typu PrintDocument**

System.Drawing.Printing.PrintDocument myPrintDocument1 = new System.Drawing.Printing.PrintDocument();

**// stworzenie obiektu typu PrintDialog (okno dialogowe)**

PrintDialog myPrinDialog1 = new PrintDialog();

**//stworzenie akcji drukowania stron**

myPrintDocument1.PrintPage += new System.Drawing.Printing.PrintPageEventHandler(myPrintDocument2\_PrintPage);

myPrinDialog1.Document = myPrintDocument1;

**// jesli zatwierdzimy okno dialogowe wykonamy wydruk**

if (myPrinDialog1.ShowDialog() == DialogResult.OK)

{

myPrintDocument1.Print();

}

}

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

{

pictureBox1.Image.Save("kod\_" + txtBarcode.Text + ".jpg"); **// zapisanie zawartosci okna do pliku jpg**

MessageBox.Show("Zapisano plik " + "kod\_" + txtBarcode.Text + ".jpg"); **// okno dialogowe informujace o zapisaniu pliku**

}  
  
**4. Wnioski**  W domu dokończyliśmy program generujący kod EAN-13, program działa poprawnie.  
Poprawność działania programu sprawdziliśmy przy pomocy jednej z aplikacji na system Android, która zczytuje kod za pomocą aparatu w smartfonie.