Paweł Twardawa 235072 Data: 23.10.2018 r.

Aleksandra Wieczorkiewicz 234980

Urządzenia peryferyjne

Ćwiczenie 2

Drukarki atramentowe. Język PCL. Drukowanie w kolorach

Prowadzący:

Dr inż. Jacek Mazurkiewicz

1. **Zadania do wykonania**
   1. Dokonać formatowania strony, czcionki i tekstu w oparciu o sekwencje języka PCL, w zakresie zmiany kroju czcionki oraz organizacji strony (plik tekstowy i wysłać do drukarki).
   2. Zdefiniować różne kolory z zakresu modeli podstawowych (RGB,CMYK) oraz palet niestandardowych, dla trybu tekstowego.
   3. Zdefiniować wybrane tryby graficzne z uwzględnieniem grafiki kolorowej.
   4. Napisać i uruchomić program, wykonujący powyższe zadania z poziomu aplikacji graficznej (sterowanie przy pomocy przycisków, list wyboru).
   5. Napisać program drukujący, wybraną w porozumieniu z prowadzącym, płaszczyznę (prostą) jednego ze wskazanych modeli składania barw.
   6. Napisać program drukujący, kolorową bitmapę w jednym z modeli składania barw.
2. **Opis programu**
   1. Przygotowanie drukarki. Przedstawiony kod na Listingu 1 odpowiedzialny jest za inicjalizacje drukarki oraz ustawienie parametrów drukowania.

Listing 1.

writer.WriteLine("\x1B" + "E"); // Reset drukarki

writer.WriteLine("\x1B" + "&s0C"); //zawijanie

writer.WriteLine("\x1B" + "&10O"); // 0 pionowo 1 poziomo

writer.WriteLine("\x1B" + "\*o0M"); // wydruk normal

writer.WriteLine("\x1B" + "&126A"); //rozmiar papieru

writer.WriteLine("\x1B" + "&a5C"); //idz do 5 kolumny

writer.WriteLine("\x1B" + "&a4R"); //idz do 4 wiersza

* 1. Formatowanie tekstu: pogrubienie, pochylenie, podkreślenie. Formatowanie tekstu zostało zrealizowanie w programie graficznym za pomocą checkbox, po zaznaczeniu odpowiedniego formatowania przesyłamy do strumienia drukarki odpowiedni kod języka PCL.

Listing 2.

if(pogrubienie.Checked)

{

writer.WriteLine("\x1B" + "(s3B");

}

if(Kursywa.Checked)

{

writer.WriteLine("\x1B" + "(s1S");

}

if(podkreslenie.Checked)

{

writer.WriteLine("\x1B" + "&d0D");

}

* 1. Rozmiar czcionki został zrealizowany za pomocą kontrolki NumericUpDown. Zakres dostępnej czcionki ograniczyliśmy od 1 do 15, gdzie 1 to najmniejsza a 15 największa. Dla drukarki im większa liczba tym mniejszy znak, aby ustawienie rozmiaru było bardziej logiczne użyliśmy prostego równania w celu zamiany wartości.   
       
     Listing 3.  
     writer.WriteLine("\x1B" + "(s"+ (numericUpDown1.Value - 16) + "H"); //zmaiana rozmiaru
  2. Do wybierania rodzaju czcionki został użyty ComboBox. Dodaliśmy 3 różne rodzaje czcionek. Niestety pomimo prób użycia różnych kodów języka PCL nie udało się zmienić rodzaju drukowanej czcionki, prawdopodobnie nie są one obsługiwane przez drukarkę.  
       
     Listing 4.

if(comboBox1.Text == "CG Times")

{

writer.WriteLine("\x1B" + "(s4101T");  
}

else if (comboBox1.Text == "Arial")

{

writer.WriteLine("\x1B" + "(s16602T");

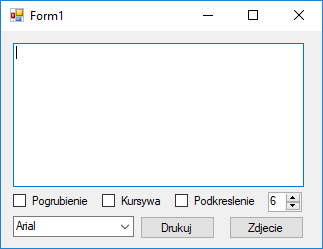
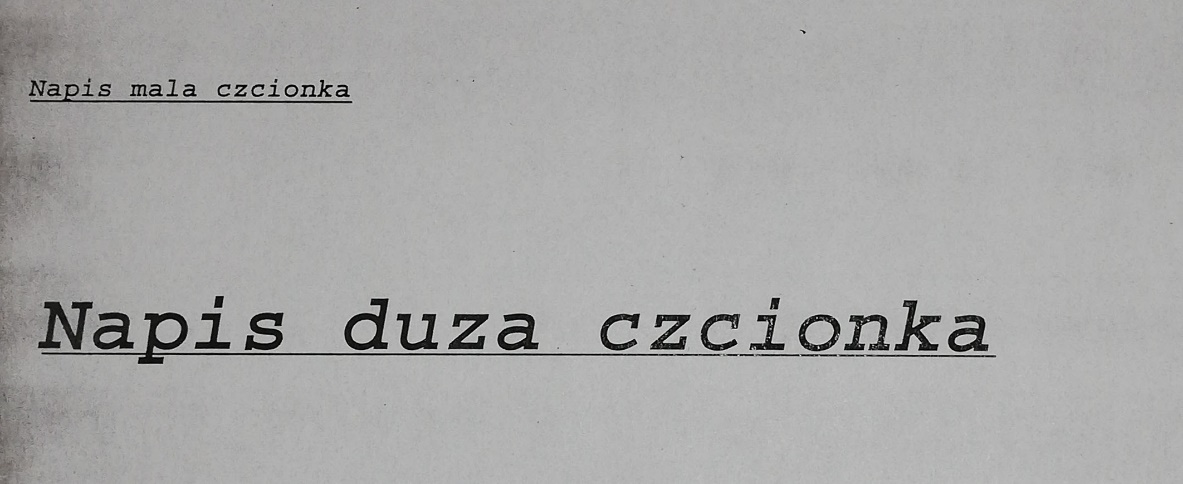
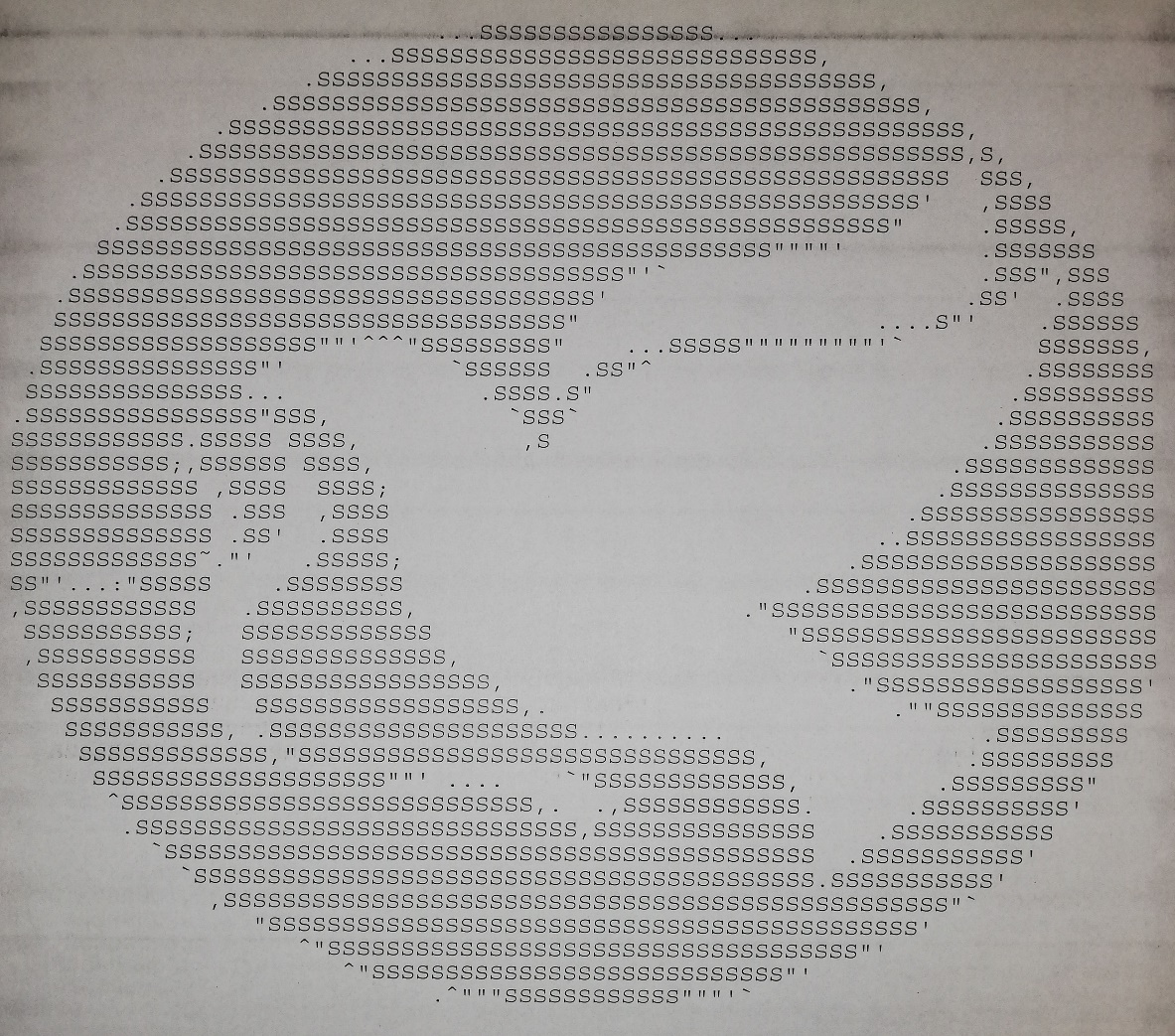
}

else if (comboBox1.Text == "Windings")

{

writer.WriteLine("\x1B" + "(s2730T");

}

* 1. Przesyłanie tekstu do drukarki odbywa się przez wpisanie go do kontrolki TextBox następnie jest on dodawany do strumienia drukarki.   
       
     Listing 5.  
     writer.WriteLine(textBox\_text.Text);
  2. Interfejs graficzny  
     
  3. Przykładowy wydruk  
     
  4. Wydruk zdjęcia

1. **Wnioski**

Język PCL został stworzony do obsługi drukarek atramentowych i laserowych. Podczas zajęć nie udało się wydrukować kolorowych napisów oraz zdjęć ponieważ w ćwiczeniu korzystaliśmy z drukarki laserowej(czarno-białej). Pozostałe zadania z powodzeniem udało się wykonać.