

Dokumentacja projektu

Języki skryptowe

Gabriela Antoniuk, grupa IA

16 grudnia 2019

Część I

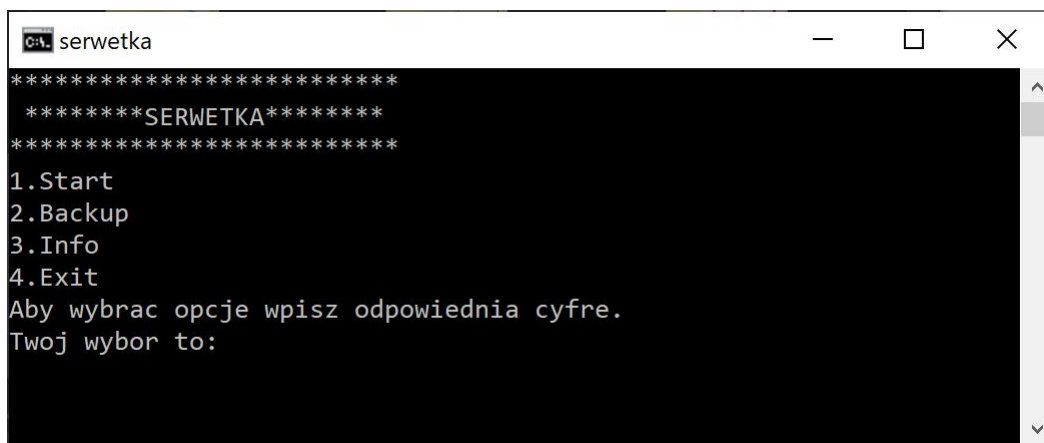
Opis programu

Program wykonuje rysunki odpowiednich krzywych zamkniętych w zależności od liczby naturalnej n .

Instrukcja obsługi

Program tworzy rysunki zależne od parametru n , zatem należy go określić. W tym celu przed uruchomieniem programu, w folderze "input" należy utworzyć plik tekstowy z parametrami n . (Uwaga! Parametry należy wpisywać jako liczby, każdy od nowej linii)

Aby uruchomić program należy włączyć plik o nazwie "start.bat". Na ekranie wyświetli się tekstowe menu z czterema pozycjami.



```
*****
*****SERWETKA*****
*****
1.Start
2.Backup
3.Info
4.Exit
Aby wybrac opcje wpisz odpowiednia cyfre.
Twój wybor to:
```

Rysunek 1: Menu programu "serwetka.bat"

Aby wybrać jedną z opcji należy wpisać liczbę przypisaną do niej (odpowiednio 1-start, 2-backup, 3-info, 4-exit) i następnie wcisnąć "Enter".

Po uruchomieniu programu i wybraniu opcji "start" program będzie tworzyć grafiki, które następnie wyświetlą się w tabeli w przeglądarce. Aby powiększyć wybraną grafikę należy na nią kliknąć. Wszystkie wyniki są zapisane w folderze /projekt/output/nazwa-folderu (gdzie "nazwa-folderu" to folder o nazwie pliku w którym zawarte były parametry).

Część II

Część techniczna

Struktura plików:

- projekt
 - input
 - * pliki wejściowe
 - output
 - * foldery z plikami wyjściowymi
 - backup
 - start.bat
 - serwetka_maker.exe
 - website_maker.py
 - index.html
 - style.css

Schemat działania

Po uruchomieniu pliku "start.bat" do wyboru są 4 opcje.

Po wybraniu pierwszej -"Start" program wyszukuje ścieżki do plików zawartych w folderze / projekt / input, następnie wywołuje plik serwetka_maker.exe z wcześniej wyszukanymi ścieżkami jako parametr.

Uruchomiony program serwetka_maker.exe korzystając z biblioteki "turtle" tworzy rysunek odpowiedni dla kolejnych wartości parametru n zawartego w pliku z folderu / projekt / input. Wyniki programu zawarte są w folderze / projekt / output / nazwa_pliku. (Gdzie nazwa_pliku to nazwa pliku który został podany jako parametry przy uruchomieniu serwetka_maker.exe)

Gdy parametr będzie nie właściwy (mniejszy bądź równy 0) w folderze / projekt / output / nazwa_pliku zostanie utworzony plik txt z odpowiednią uwagą. Dla pozostałych dodatnich parametrów z wykorzystaniem Ghostscript zostanie utworzony plik jpg z rysunkiem o nazwie wartości parametru. Po zakończeniu pracy programu uruchomi się plik index.html w którym zawarte będą wyniki w formie tabeli.

Po wybraniu drugiej opcji-"Backup" programu "start.bat" zostanie wykonane kopiowanie wszystkich plików zawartych w folderze / projekt do uprzednio utworzonego folderu "backup".

Po wybraniu trzeciej opcji-"Info" programu "start.bat" na ekranie pojawi się informacja jak korzystać z programu. Po wybraniu czwartej opcji-"Exit" program zostanie zamknięty.

Opis działania

Zawarty w pliku "serwetka_maker.exe" algorytm wykorzystuje zależność parametru n od ilości rysowanych kresek.

$$ilosc_kresiek(n) = (n - 1) * 16 + 8$$

Implementacja

Data: zakręty_w_prawo, ilość_zakrętów, n, ilosc_kreszek

Result: Rysunek

zakręty_w_prawo=0;

ilość_zakrętów=0;

for $j = n$; $j > 0$; $j = j - 1$ **do**

for $i = 0$; $i == ilosc_kreszek$; $i = i + 1$ **do**

if i jest parzyste **then**

 Skręć w prawo; Rysuj kreskę; **if** $zakręty_w_prawo = 2*j - 1$ **then**

 Skręć w prawo;

 Rysuj kreskę;

 ilość_zakrętów+=1;

 zakręty_w_prawo=1;

else

 zakręty_w_prawo+=1;

 ilość_zakrętów+=1;

end

else

 ilość_zakrętów dodaj+=1;

 Skręć w lewo;

 Rysuj kreskę;

end

end

 Skręć w prawo;

 Dwa razy idź do tyłu;

 Skręć w lewo;

end

Algorithm 1: Algorytm rysowania serwetki

Pełen kod programu

Wklejamy pełen kod z podziałem na pliki, np.:

- start.bat

```
@echo off
setlocal enabledelayedexpansion
title serwetka
:menu
cls
echo *****
echo  *****SERWETKA*****
echo *****
echo 1.Start
echo 2.Backup
echo 3.Info
echo 4.Exit

echo Aby wybrac opcje wpisz odpowiednia cyfre.
set /p wybor="Twoj wybor to: "
if %wybor%==1 goto m_start
if %wybor%==2 goto backup
if %wybor%==3 goto info
if %wybor%==4 goto exit
goto error

:m_start
cls
forfiles /p .\input /C "cmd /c ..\serwetka_maker.exe @file"
set filename=
set date= %date% %time%
for %%a in (.\input\*.txt) do (
    set filename=%%~nxa
    call set filename=!filename:.txt=!
    set files=
    for %%i in ("output\!filename!\*") do (
        call set files=!files! "%%i"
    )
    website_maker.py "%%a" "!date!"!files!
)
index.html
goto exit

:backup
cls
```

```

if not exist backup mkdir backup
@ROBOCOPY ..\projekt .\backup /S /XD backup
cls
echo Backup wykonany pomyslnie
set /p enter="Wcisnij Enter aby kontynuowac"
goto exit

:info
cls
echo Serwetka - info
echo Program tworzy obrazy o rozszerzeniu JPG na ktorych znajduje sie serwetka
o wielkosci zaleznej od zadanego parametru.
echo.
echo INSTRUKCJA
echo 1. W folderze "input" umiesc plik tekstowy z parametrami (kazdy w nowej
linii).
echo 2. Uruchom start.bat
echo 3. Wybierz pierwsza opcje (1.Start)
echo 4. Gotowe
echo.
echo Utworzone obrazy znajduja sie w folderze "output".
set /p enter="Wcisnij Enter aby kontynuowac"
goto menu

:error
echo Zly wybor
set /p enter="Wcisnij Enter aby kontynuowac"
goto menu

:exit
echo on

```

- serwetka_maker.exe

```

import turtle
import io
from PIL import Image
import sys
import os

class Serwetka:
    def __init__(self, m, filename):
        self.name=filename
        if(m>0):
            self.n=m
        else:
            self.n=m

```

```

        if not os.path.exists('..\output\\' + self.name):
            os.mkdir('..\output\\' + self.name)
        f=open('..\output\\' + self.name + '\\'+ str(self.n) + '.txt', "a")
        f.write("Podano niewlasciwy parametr: "+str(m))
        f.close()
    self.przod=5
    self.zakret=90
    self.zakrety=1
    self.zakretywprawo=1
    self.height=2*self.n*self.przod+self.przod

def __zolw__(self):
    self.zolw = turtle.Turtle()
    self.__window__()
    self.zolw.hideturtle()
    self.zolw.speed(0)
    self.zolw._tracer(0,0)
    self.zolw.up()
    self.zolw.setpos(-turtle.window_width()/2+self.height+self.przod,
                    turtle.window_height()/2-2*self.przod)
    self.zolw.down()

def __window__(self):
    self.window=self.zolw.getscreen()
    self.window.update()
    self.canva=self.window.getcanvas()

def makeSerwetka(self):
    if(self.n>0):
        self.__zolw__()
        for j in range(self.n,0,-1):
            kreski=(j-1)*16+8
            for i in range(kreski):
                if(i%2==0):
                    self.zolw.right(self.zakret)
                    self.zolw.forward(self.przod)
                    if(self.zakretywprawo+1==2*j):
                        self.zolw.right(self.zakret)
                        self.zolw.forward(self.przod)
                        self.zakrety+=1
                        self.zakretywprawo=1
                    else:
                        self.zakretywprawo+=1
                        self.zakrety+=1
            else:
                self.zakretywprawo+=1
                self.zakrety+=1
        else:
            self.zakretywprawo+=1
            self.zakrety+=1
    else:

```

```

        self.zakrety+=1
        self.zolw.left(self.zakret)
        self.zolw.forward(self.przod)
    self.zolw.up()
    self.zolw.right(self.zakret)
    self.zolw.forward(2*self.przod)
    self.zolw.left(self.zakret)
    self.zolw.down()
self.canva.update()
self.save()
self.window.clear()

def save(self):
    #self.name += '_' +
    ps = self.canva.postscript(width=2*self.height,height=2*self.height)
    im = Image.open(io.BytesIO(ps.encode('utf-8')))
    if not os.path.exists('..\output\\' + self.name):
        os.makedirs('..\output\\' + self.name)
    im.save('..\output\\' + self.name + '\\'+ str(self.n) +'.jpg')

class Parametr:
    def __init__(self, file):
        self.f=open(file,'r')
        if(self.f.mode=='r'):
            self.name=os.path.basename(self.f.name).replace(".txt","")
        else:
            if not os.path.exists('..\output\\' + self.name):
                os.makedirs('..\output\\' + self.name)
            f=open('..\output\\'+self.name+'\\'+self.name+'_error.txt','a')
            f.write("Nie udało się otworzyć pliku "+self.name)
            f.close()

path=str(sys.argv[1])
parametr=Parametr(path)

for line in parametr.f:
    serwetka=Serwetka(int(line),parametr.name)
    serwetka.makeSerwetka()

```

- website_maker.py

```

import sys
import os
path='index.html'
f=open('index.html','r+')
content=f.readlines()

```



```

f.close()
f=open('index.html','w+')

filename=sys.argv[1]
f1=open(filename,'r')
parameters=""
for p in f1:
    parameters+=p.replace('\n',',', ' ')

filename=os.path.basename(filename)
date=sys.argv[2]

files=""
for pic in range(3,len(sys.argv)):
    if (str(sys.argv[pic]).find('.jpg')!=-1):
        files+='<a target=\'_blank\' href=\''+str(sys.argv[pic])+\'\'>
            <img height=\'100\' src=\''+str(sys.argv[pic])+\'\'></a>\n'
    else:
        files+='<a target=\'_blank\' href=\''+str(sys.argv[pic])+\'\'>
            +os.path.basename(str(sys.argv[pic]))+\'</a>'
newContent='<tr><td>'+date+'</td><td>'+filename+'</td><td>'+parameters
            +\'</td><td>'+files+\'</td></tr>\n</table>\n'

i=0
for line in content:
    if line=='</table>\n':
        content[i]=newContent
    i+=1
f.writelines(content)
f.close()
f1.close()

```