

# Dokumentacja projektu

## Języki Skryptowe

Mikołaj Pazera, grupa 1B

21 stycznia 2020

# Część I

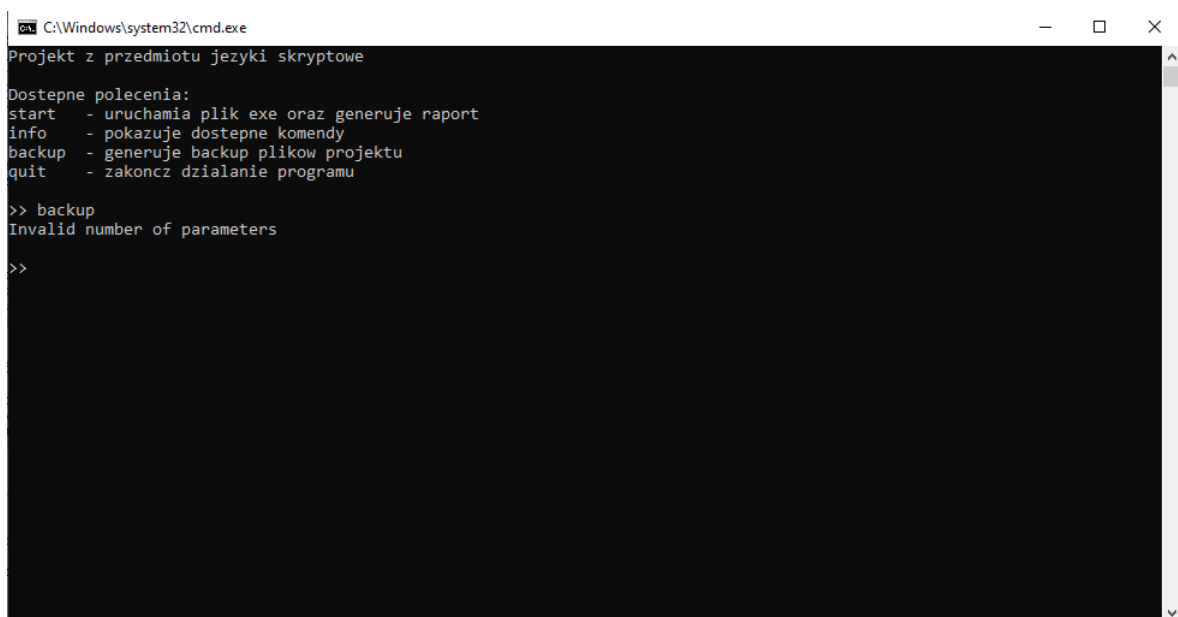
## Opis programu

Program służy do sprawdzania czy dana para tworzy tzw. liczby zaprzyjaźnione. Dane pobierane są z plików .txt. Polecenie:

Za Wikipedią: *Liczby zaprzyjaźnione to para różnych liczb naturalnych, takich, że suma dzielników każdej z tych liczb równa się drugiej (nie uwzględniając tych dwóch liczb jako dzielników).*" Np. liczba 284 ma dzielniki: 1, 2, 4, 71, 142, których suma daje 220, a liczba 220 ma dzielniki: 1, 2, 4, 5, 10, 11, 20, 22, 44, 55, 110, których suma daje 284. Zatem liczby 220 i 284 tworzą parę liczb zaprzyjaźnionych. Należy napisać program, który dla dowolnej pary różnych liczb naturalnych będzie rozstrzygał, czy para ta tworzy liczby zaprzyjaźnione.

## Instrukcja obsługi

Aby uruchomić program należy otworzyć plik start.bat znajdujący się w głównym katalogu projektu. Wyświetli się menu programu:



```
C:\Windows\system32\cmd.exe
Projekt z przedmiotu języki skryptowe

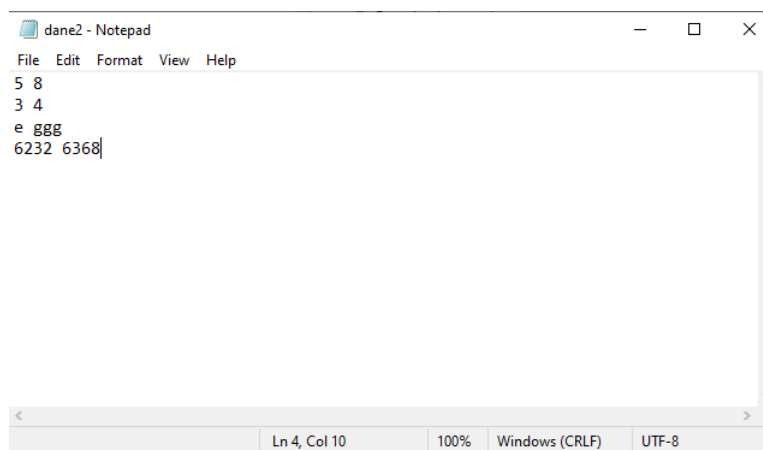
Dostępne polecenia:
start - uruchamia plik exe oraz generuje raport
info - pokazuje dostępne komendy
backup - generuje backup plików projektu
quit - zakończ działanie programu

>> backup
Invalid number of parameters

>>
```

Program pobiera dane od użytkownika z pliku tekstowego. Pary liczb są sprawdzane a następnie generowany jest raport .html. Dane niepoprawne zostaną uwzględnione w raporcie jako błędy. Dane do programu należy umieścić w folderze daneWejsciowe w plikach .txt

Przykładowa zawartość pliku .txt:



```
dane2 - Notepad
File Edit Format View Help
5 8
3 4
e ggg
6232 6368
Ln 4, Col 10 100% Windows (CRLF) UTF-8
```

Aby uruchomić program należy użyć polecenia `start`. Inne dostępne polecenia to:

- `backup` - dokonuje kopii zapasowej wszystkich plików projektu
- `info` - wyświetla informacje dotyczące działania programu
- `exit` - kończy działanie projektu

## Część II

### Część techniczna

Program składa się z trzech głównych elementów: skrypt batch odpowiedzialny za uruchamianie kolejnych elementów programu, plik `.exe` sprawdzający poprawność danych wejściowych i sprawdzający warunki zadania oraz z pliku python który odpowiada za przeniesienie wyników do raportu `.html`. Wykorzystana wersja Pythona to 3.8. Plik wykonywalny `python.exe` musi być dodany do ścieżki systemowej w celu poprawnego działania programu

### Opis działania

Sprawdzanie czy para liczb tworzy liczby zaprzyjznione odbywa się za pomocą szukania dzielników obu liczb. Nie jest używany w tym celu konkretny wzór lecz sprawdzanie wszystkich liczb  $i$  od 0 do  $n$ . Jeśli  $i$  jest dzielnikiem liczby  $n$  to dodawana jest ona do sumy dzielników

$$i \bmod (n) \equiv 0 \quad (1)$$

Jesli suma dzielników liczby (z wyłączeniem samej liczby)  $a$  jest równa liczbie  $b$  (i odwrotnie) to liczby tworzą parę liczb zaprzyjaznionych. Przykładowo sprawdzenie liczb 220 oraz 286:

Dzielniki liczby <b>220</b>	Dzielniki liczby <b>284</b>
1	1
2	2
4	4
71	5
142	10
	11
	20
	22
	44
	55
	110
Suma: <b>284</b>	Suma: <b>220</b>

## Pseudokod

**Result:** Program Liczby Zaprzyjznione

Wczytaj wybor uzytkownika;

start;

**if** *wybor = start* **then**

    otworz plik .exe;

    pobierz dane;

    sprawdz czy sa poprawne;

    dla poprawnych danych sprawdz warunek zadania;

    zapisz informacje do pliku wyjsciowego;

    otworz plik .py;

    pobierz dane z plikow wyjsciowych;

    zapisz je do pliku .html;

    otworz plik .html;

**end**

**if** *wybor = backup* **then**

    kopiuj kazdy podfolder projektu do folderu backup;

**end**

**if** *wybor = info* **then**

    wyswietl informacje o projekcie;

**end**

**if** *wybor = exit* **then**

    zakoncz program;

**end**

**Algorithm 1:** Pseudokod programu

## Pełny kod programu

Listing 1: Plik start.bat

```
@echo off

echo Projekt z przedmiotu jezyki skryptowe
echo.
:info
echo Dostepne polecenia:
echo start      - uruchamia plik exe oraz generuje raport
echo info      - pokazuje dostepne komendy
echo backup     - generuje backup plikow projektu
echo quit      - zakoncz dzialanie programu

:wybor
echo.
set /p wybor=">> "
if \%wybor\%==start goto start
if \%wybor\%==info goto info
if \%wybor\%==backup goto backup
if \%wybor\%==quit goto quit
echo Niepoprawne polecenie
goto wybor

:start
echo.
del /q daneWyjscowe\*
for /r \%v in (daneWejscowe\*.txt) do call aplikaca_exe\Debug\aplikaca_exe.exe "
del /q index.html
for /r \%v in (daneWyjscowe\*.txt) do call python.exe python_raport\python_rapor
call index.html
goto wybor

:backup
if not exist backup (mkdir backup)
xcopy /q /y \%cd%\index.html \%cd%\backup
goto wybor

:quit
pause
echo on
```

Listing 2: Plik start.bat

```
#include <iostream>
#include <string>
#include <fstream>
#include <sstream>

using namespace std;
int wczytajLiczby(string);
void sprawdzCzyZaprzyjaznione(int, int, string);
void zapiszDoPliku(int, int, bool, bool, string);

int main(int argc, char* argv[])
{
    string nazwaPliku = "";
    nazwaPliku += argv[1];

    wczytajLiczby(nazwaPliku);
    return 0;
}

int wczytajLiczby(string nazwa) {
    ifstream plikWejsciowy(nazwa);
    string wiersz;
    while (getline(plikWejsciowy, wiersz))
    {
        istringstream iss(wiersz);
        int a, b;
        if (!(iss >> a >> b)) {
            zapiszDoPliku(0, 0, false, true, nazwa);
        }
        else {
            sprawdzCzyZaprzyjaznione(a, b, nazwa);
        }
    }
    plikWejsciowy.close();
    return 0;
}

void sprawdzCzyZaprzyjaznione(int liczbaA, int liczbaB, string nazwa) {
    int sumaDzielnikowA = 0, sumaDzielnikowB = 0;
    for (int i = 1; i < liczbaA; i++) {
        if (liczbaA % i == 0) {
            sumaDzielnikowA += i;
        }
    }
    for (int i = 1; i < liczbaB; i++) {
        if (liczbaB % i == 0) {
            sumaDzielnikowB += i;
        }
    }
    if (sumaDzielnikowA == sumaDzielnikowB) {
        zapiszDoPliku(liczbaA, liczbaB, true, false, nazwa);
    }
}
```

```

        }
    }
    for (int i = 1; i < liczbaB; i++) {
        if (liczbaB % i == 0) {
            sumaDzielnikowB += i;
        }
    }
    if (sumaDzielnikowA == liczbaB && sumaDzielnikowB == liczbaA) zapiszDoPliku
    else zapiszDoPliku(liczbaA, liczbaB, false, false, nazwa);
}

```

```

void zapiszDoPliku(int a, int b, bool czyZaprzyjzaznione, bool czyError, string nazwa

```

```

    ofstream plikWyjsciowy;
    string toReplace("daneWejsciowe");
    size_t pos = nazwa.find(toReplace);
    nazwa.replace(pos, toReplace.length(), "daneWyjsciowe");
    plikWyjsciowy.open(nazwa, ios_base::app);

    if (czyError) {
        plikWyjsciowy << "b" << endl;
    }
    else {
        plikWyjsciowy << a << " ";
        plikWyjsciowy << b << " ";
        if(czyZaprzyjzaznione) plikWyjsciowy << "t" << endl;
        else plikWyjsciowy << "n" << endl;
    }

    plikWyjsciowy.close();
}

```

Listing 3: Plik python\_raport.py

```

import sys
import os
nazwa = str(sys.argv[1])
plikWyjsciowy = open("index.html", "a+")
template = ""
if(os.stat("index.html").st_size == 0):
    template = """
<html>
<head>
<title>Raport</title>

```



```

<link rel="stylesheet" href="styl.css">
<style>
body{
background: #a0d8ef;
background: -moz-linear-gradient(top,  #a0d8ef 0%, #ddffff 80%);
background: -webkit-linear-gradient(top,  #a0d8ef 0%,#ddffff 80%);
background: linear-gradient(to bottom,  #a0d8ef 0%,#ddffff 80%);
filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#a0d8ef', endColorstr='#ddffff',gradientType=1);
}

*{
font-family: "Arial";
}
table{

margin: auto;
margin-top: 20px;
border-collapse: collapse;
}

.lewa, .t-lewa{
text-align: right;
padding-right: 10px;
border-right: 1px solid #8c8c8c;
padding-top: 3px;
}

.prawa, .t-prawa{
text-align: left;
padding-left: 10px;
border-left: 1px solid #8c8c8c;
padding-top: 3px;
}

.t-lewa, .t-prawa{
border-bottom: 1px solid #8c8c8c;
padding-bottom: 10px;
}

.suma{
border-top: 1px solid #8c8c8c;
}

```

```
.true, .false, .error {
font-weight: 700;
}
```

```
.true{
color: #2d862d;
}
```

```
.false{
color: #b30000;
}
```

```
</style>
```

```
</head>
```

```
<body>"""
```

```
plikWejscowy = open(nazwa, "r")
```

```
template += "<p>Plik: "+str(nazwa)+"</p>"
```

```
wiersz = plikWejscowy.readlines()
```

```
blad = 0
```

```
zapr = 0
```

```
niezap = 0
```

```
for linia in wiersz:
```

```
    if("b" in linia):
```

```
        template += ""
```

```
        <p class="error">Wiersz w pliku wejscowym zawiera bledne dane</p>
        ""
```

```
        blad += 1
```

```
    else:
```

```
        liczby = linia.split()
```

```
        if(liczby[2] == "t"):
```

```
            template += ""
```

```
            <p class="true"> Liczby "" + str(liczby[0]) + "" oraz "" + str(liczb
            ""
```

```
            zapr += 1
```

```
        else:
```

```
            template += ""
```

```
            <p class="false"> Liczby "" + str(liczby[0]) + "" oraz "" + str(lic
            ""
```

```
            niezap += 1
```

```
template += "<p>Ilosc blednych danych w pliku: " + str(blad) + "</p>"
```

```
template += "<p>Ilosc par zaprzyjaznionych: " + str(zapr) + "</p>"
```

```
template += "<p>Ilosc par niezaprzyjaznionych: " + str(niezap) + "</p>"
```

```
plikWyjsciowy.write(template)
plikWejsciowy.close()
plikWyjsciowy.close()
```