

# Dokumentacja Projektu

## Program Liczb Zaprzyjaźnionych

Natalia Szalas  
Informatyka, sem. III, gr. 2D  
Wydział Matematyki Stosowanej

15 Stycznia 2021

# Spis treści

<b>1</b>	<b>Część I</b>	<b>3</b>
1.1	Treść zadania: Liczby zaprzyjaźnione . . . . .	3
1.2	Opis programu . . . . .	3
1.3	Pseudokod . . . . .	3
1.4	Instrukcja obsługi i działania . . . . .	4
1.5	Schematy blokowe wybranych funkcji . . . . .	7
<b>2</b>	<b>Część II</b>	<b>11</b>
2.1	Część techniczna . . . . .	11
2.2	PLIKI PROJEKTU . . . . .	11
<b>3</b>	<b>PODSUMOWANIE</b>	<b>12</b>
3.1	Raport ogólny . . . . .	12
3.2	Propozycje ulepszeń . . . . .	12

# 1 Część I

## 1.1 Treść zadania: Liczby zaprzyjaźnione

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.

## 1.2 Opis programu

Program ma na celu ustalenie czy dwie liczby są zaprzyjaźnione. Oznacza to, że suma wszystkich dzielników pierwszej liczby, musi się równać drugiej liczbie. Przykład: jeśli mamy liczby 220 oraz 284:

- $220 = 1 + 2 + 4 + 71 + 142$  (dzielniki 284)
- $284 = 1 + 2 + 4 + 5 + 10 + 11 + 20 + 22 + 44 + 55 + 110$  (dzielniki 220)

## 1.3 Pseudokod

**Data:** num1, num2, areFriends(), sumDivisors()

**Result:** Sprawdzenie czy liczby są zaprzyjaźnione

funkcja areFriends();

**if** *num1*  $\neq$  *num2* **then**

**if** *num1* == *sumDivisors*(*num2*) *num2* == *sumDivisors*(*num1*) **then**

        | liczby są zaprzyjaźnione

**else**

        | liczby nie są zaprzyjaźnione

**end**

    ;

**else**

    | numery są takie same, więc nie można ich porównać

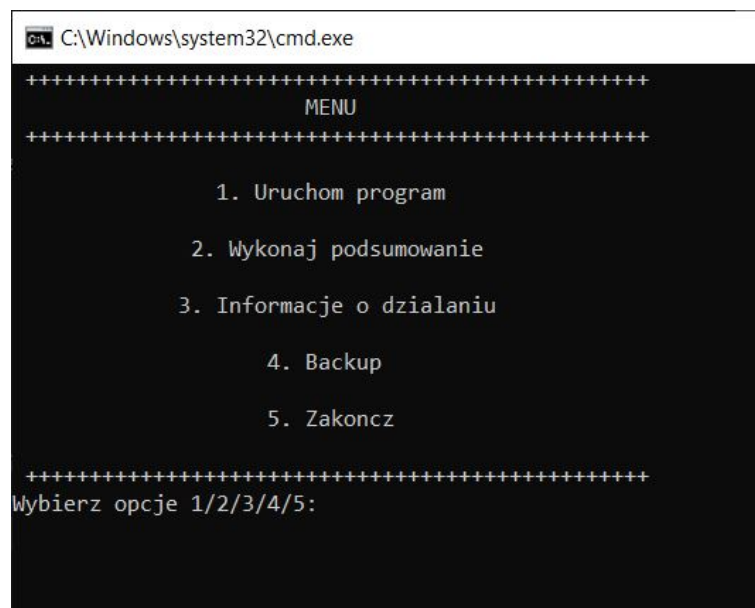
**end**

;

## 1.4 Instrukcja obsługi i działania

Po uruchomieniu aplikacji pojawia się menu z opcjami do wyboru:

1. Uruchom program - uruchamia plik .py, zwraca 2 pliki .txt
2. Wykonaj podsumowanie - uruchamia plik .py, zwraca html
3. Informacje o działaniu
4. Backup - tworzy kopię zapasową
5. Zakoncz - kończy prace programu



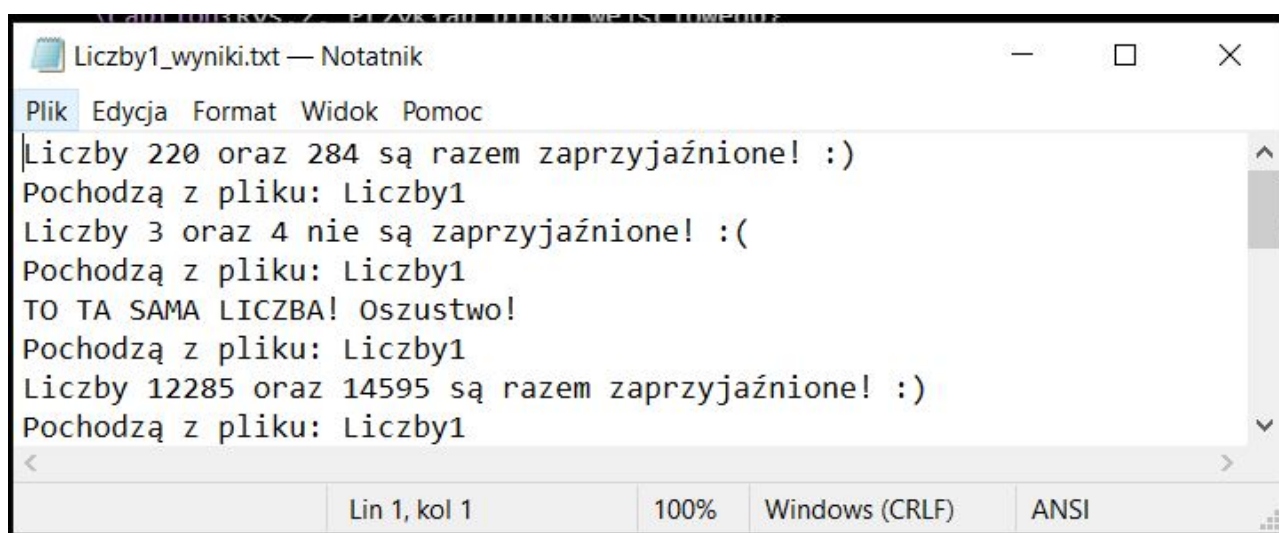
Rysunek 1: Przykład menu

W celu uzyskania rezultatów, należy najpierw odpowiednio napisać plik tekstowy. Oznacza to, że każda para liczb musi znajdować się w osobnej linijce. Do tego trzeba oddzielić je spacją oraz dodać spację za drugą liczbą. (rys.2)

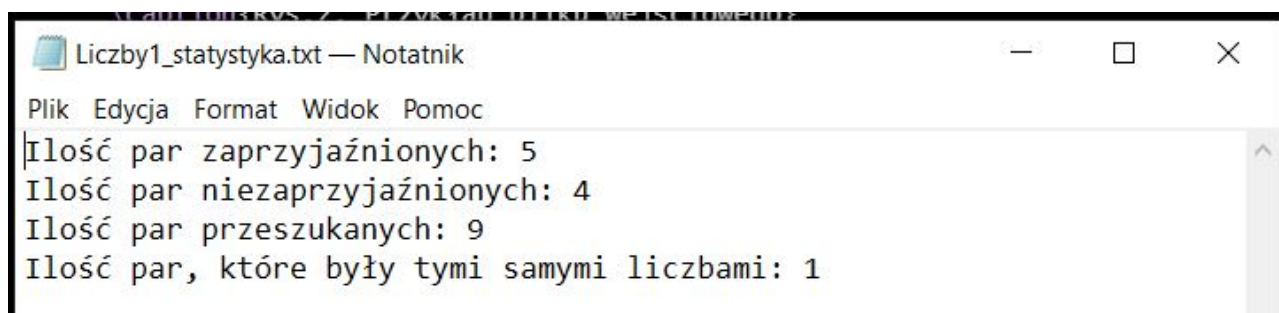


Rysunek 2: Przykład pliku wejściowego

Przy wyborze opcji numer 1, po wpisaniu nazwy pliku, rozpocznie się proces sprawdzania czy liczby są zaprzyjaźnione, a jego rezultaty zostaną zapisane do pliku tekstowego(rys.3). W tym samym czasie wykona się statystyka, która również pojawi się w pliku txt(rys.4).



Rysunek 3: Plik tekstowy z wynikami



Rysunek 4: Plik tekstowy ze statystyką

Przy opcji numer 2, po wpisaniu odpowiedniej nazwy, rozpocznie się przetwarzanie danych z wyniki.txt oraz statystyka.txt, w wyniku którego utworzy się strona HTML z jasnym raportem na temat interesującego nas tematu(rys.5).



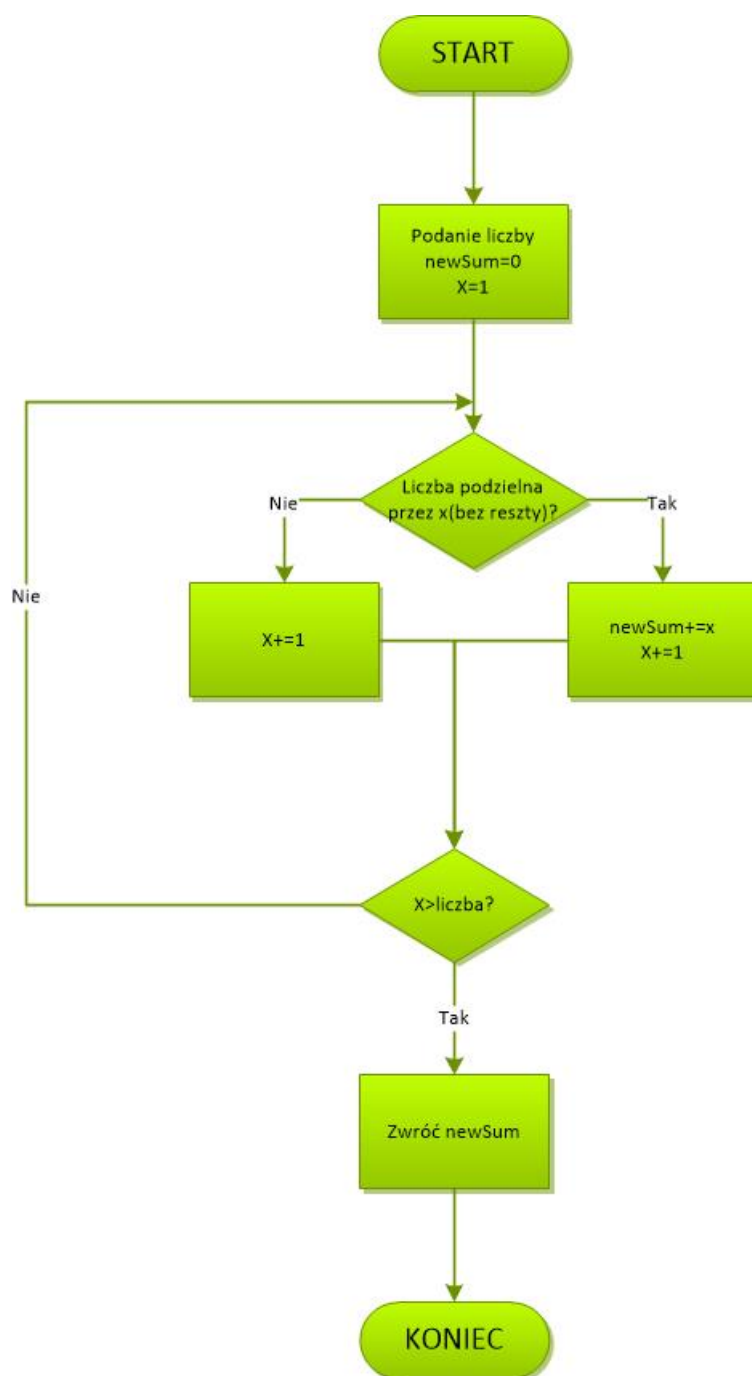
Rysunek 5: Strona HTML

Przy opcji numer 3, otworzą się informacje odnośnie programu.

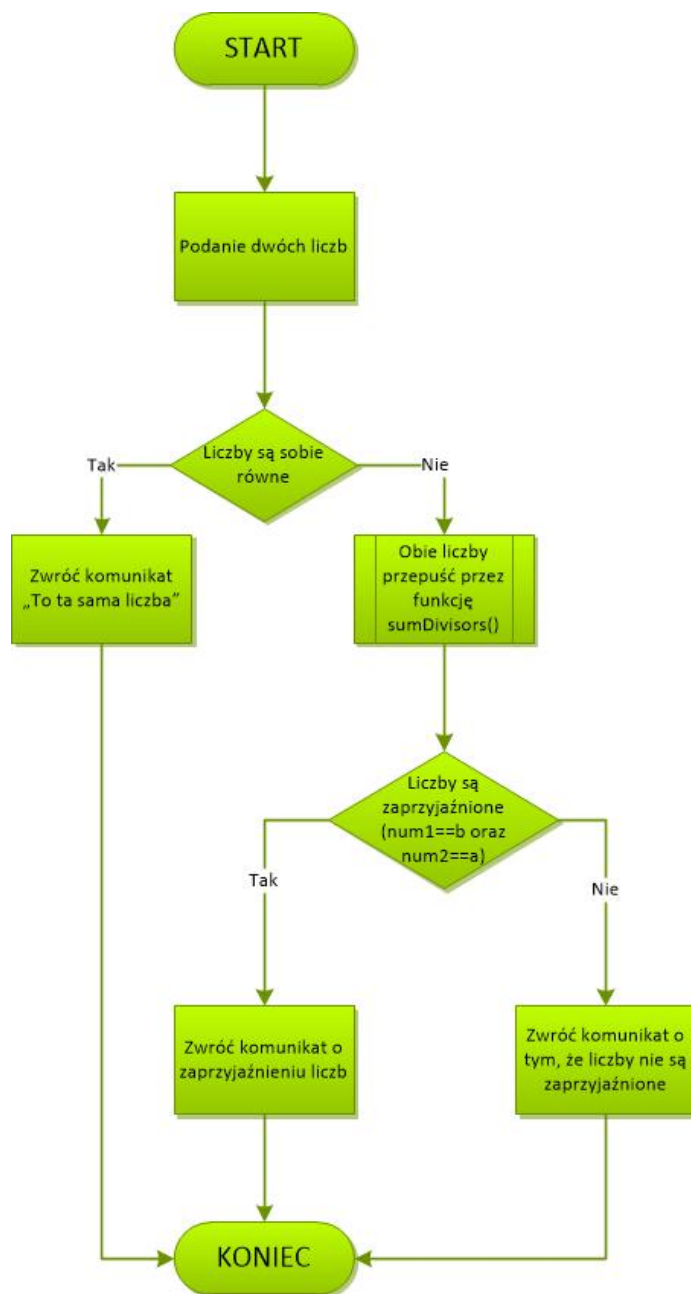
Przy opcji numer 4 utworzy się kopia zapasowa w osobnym pliku zwanym backup.

Opcja numer 5 zakończy działanie programu i zamknie okienko.

## 1.5 Schematy blokowe wybranych funkcji

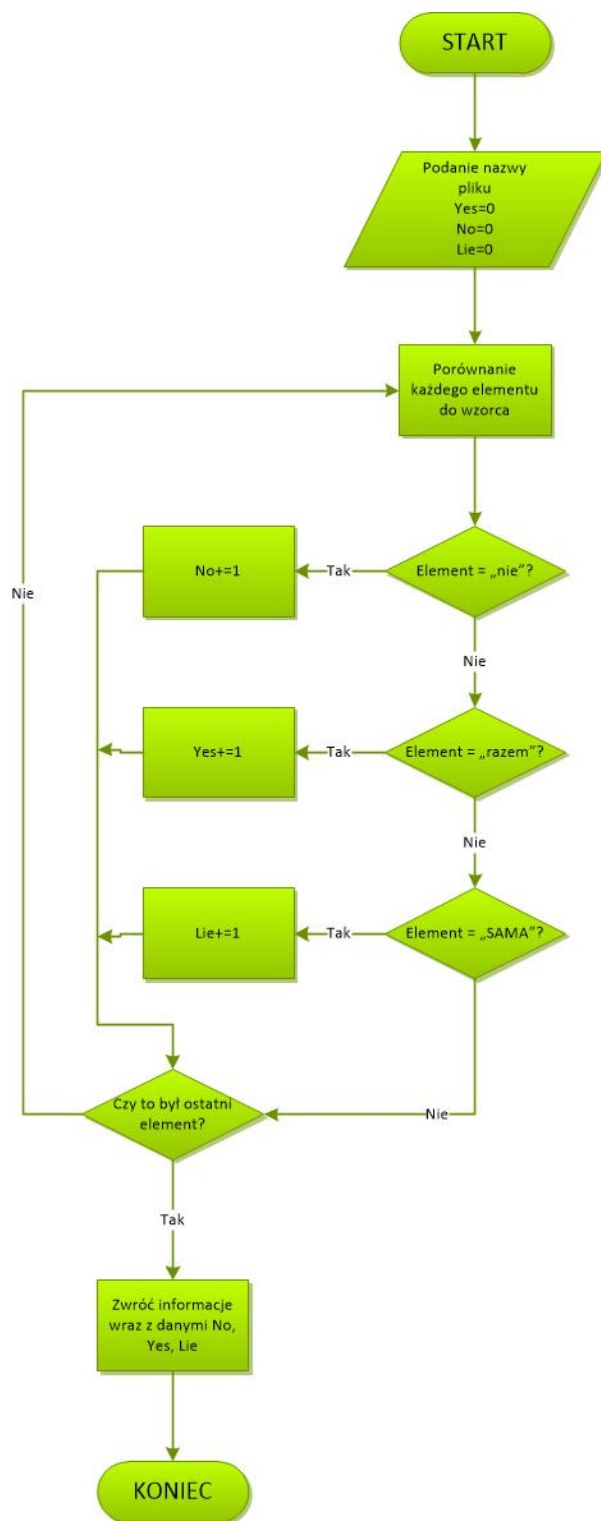


Rysunek 6: Funkcja `sumDivisors()`

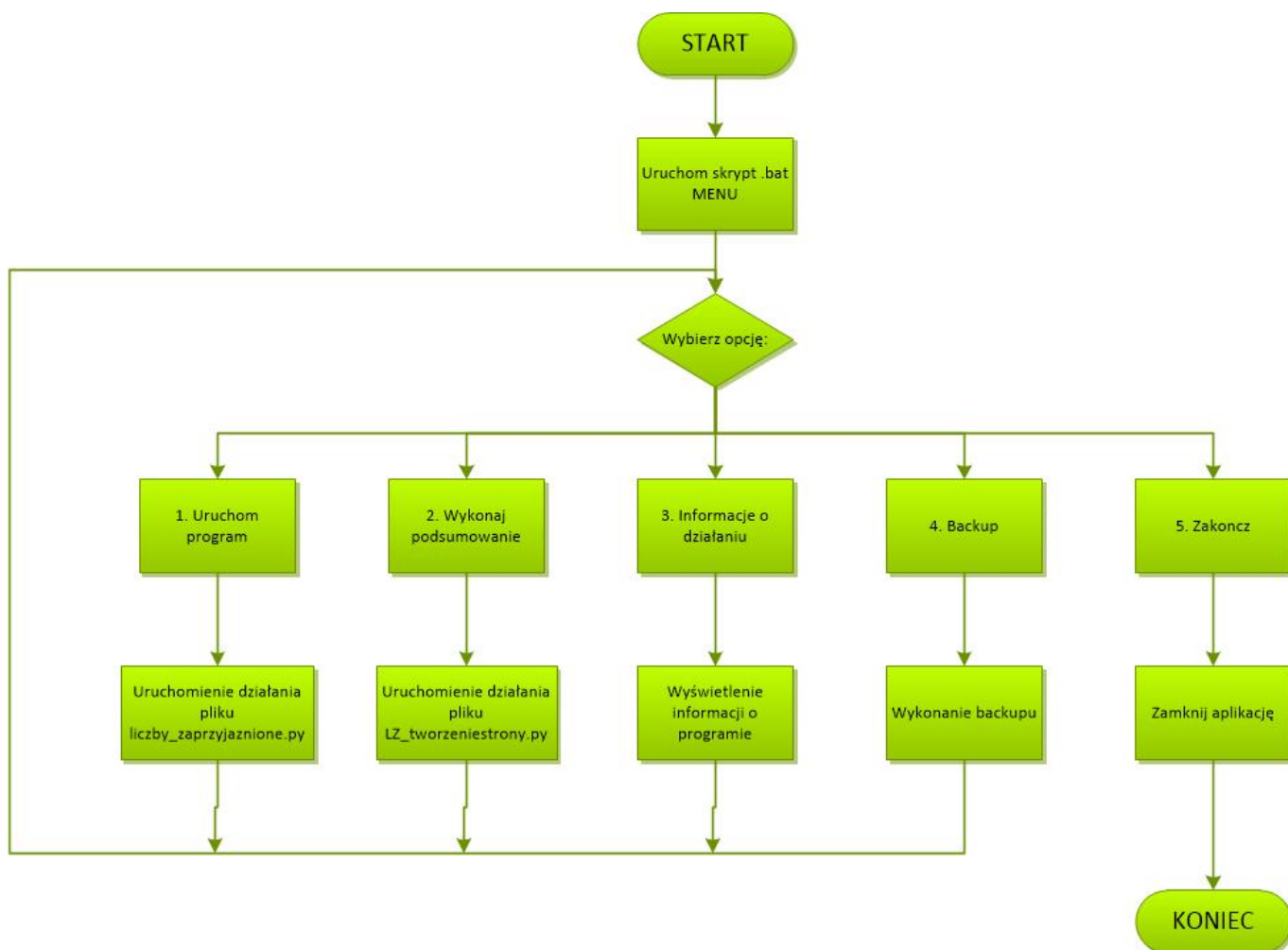


Rysunek 7: Funkcja `areFriends()`





Rysunek 8: Funkcja statistics()



Rysunek 2: Schemat działania Menu z pliku .bat

## 2 Część II

### 2.1 Część techniczna

Program składa się z plików: .bat, .py, .txt oraz .html. Po uruchomieniu pliku bat, wchodzimy w opcję pierwszą. Wówczas uruchamia się plik liczby\_zaprzyjaznione.py, zawierający cały algorytm:

Po wprowadzeniu nazwy pliku do sprawdzenia, uruchamia się funkcja LoadFile() wyszukująca danych do pobrania. Gdy operacja przejdzie pomyślnie, dane są dzielone w odpowiedni sposób do dalszej obróbki. Następnie uruchamia się funkcja areFriends(), w której na podanych dwóch liczbach sprawdza się warunki bycia zaprzyjaznionymi. W środku znajdziemy również funkcję sumDivisors(), która w szybki sposób oblicza sumę dzielników danej liczby. Później uzyskane rezultaty zapisują się do pliku tekstowego za pomocą funkcji SaveFile(). Na sam koniec tworzy się również plik statystyk zliczający ilość odpowiednich słów i zwracający uzyskane rachunki w postaci pliku txt.

Kolejnym procesem jest pełen raport uzyskany z opcji numer dwa w menu, który analizując pliki txt, zwraca plik html.

### 2.2 PLIKI PROJEKTU

- Projekt
  - liczby\_zaprzyjaznione.bat
  - liczby\_zaprzyjaznione.py
  - Liczby1.txt
  - Liczby1\_statystyka.txt
  - Liczby1\_wyniki.txt
  - Liczby1\_zaprzyjaznione.html
  - Liczby2.txt
  - LZ\_tworzeniestrony.py
- Dokumentacja
  - Dokumentacja.pdf
  - Grafiki(folder)
- backup

## 3 PODSUMOWANIE

### 3.1 Raport ogólny

Program działa poprawnie i zgodnie z przeznaczeniem. Został napisany w systemie operacyjnym Microsoft Windows 10 Home, a jego część python'owska w Visual Studio Code. Aplikacja nie była testowana w systemie Linux.

Zadanie z powyższym algorytmem pochodzi z konkursu 'Algorytmion' z roku 2013.

Nie udało się zrealizować odczytu plików z osobnego folderu, ani zapisu plików do innego folderu niż ten, w którym znajdują się wszystkie pliki .bat oraz .py.

### 3.2 Propozycje ulepszeń

Program nie został napisany w celach komercyjnych, jednak gdyby chcieć go sprzedać, należałoby popracować nad oprawą graficzną. Można również stworzyć ulepszenia w postaci automatycznego backupu co miesiąc oraz dodatkowych funkcji, bądź dopisania kilku innych algorytmów. W ten sposób powstałaby aplikacja Albus, która miałaby więcej niż jeden cel. Warto by również umniejszyć wkład użytkownika, w intencji wykazywania mniejszej ilości błędów.

## PEŁEN KOD PROGRAMU

- liczby\_zaprzyjzaznione.bat

```
1          @echo off
2
3 goto MENU
4 :MENU
5 cls
6 echo      ++++++
7 echo      MENU
8 echo      ++++++
9 echo.
10 echo      1. Uruchom program
11 echo.
12 echo      2. Wykonaj podsumowanie
13 echo.
14 echo      3. Informacje o dzialaniu
15 echo.
16 echo      4. Backup
17 echo.
18 echo      5. Zakoncz
19 echo.
20 echo      ++++++
21
22 set /p wybor="Wybierz opcje 1/2/3/4/5: "
23 If %wybor%==1 goto URUCHOM
24 If %wybor%==2 goto ZACZNIJ
25 If %wybor%==3 goto INFO
26 If %wybor%==4 goto BACKUP
27 If %wybor%==5 goto exit
28
29 :URUCHOM
30 cls
31 liczby_zaprzyjzaznione.py
32 set /p wybor="Aby wrocic do MENU nacisnij ENTER..."
33 goto MENU
34
35 :ZACZNIJ
36 cls
37 LZ_tworzenieistrony.py
38 set /p wybor="Aby wrocic do MENU nacisnij ENTER..."
39 goto MENU
40
41 :INFO
42 cls
43 echo      ++++++
44 echo      INFORMACJE O PROGRAMIE
45 echo      ++++++
46 echo.
47 echo      Glownym celem programu jest sprawdzenie czy
48 echo      suma wszystkich dzielnikow pierwszej z liczb
49 echo      rowna sie liczbie drugiej i na odwrot.
50 echo.
51 echo      Aby program poprawnie zadzialal, kazda para
```

```

52 echo      liczb musi znajdowac sie w osobnej linijce pliku
53 echo      tekstowego. Do tego liczby musza zostac
54 echo      oddzielone spacja, ktora musi tez zostac dodana
55 echo      za druga z nich.
56 echo.
57 echo      Istnieje rowniez funkcja "Wykonaj podsumowanie"
58 echo      wyswietlajaca wyniki dla wczesniej uruchomionego
59 echo      programu w postaci strony HTML.
60 echo.
61 echo.
62 echo      Program napisany przez Natalie Szalas.
63 echo.
64 echo.
65 set /p wybor="Aby wrocic do MENU nacisnij ENTER..."
66 goto MENU
67
68
69 :BACKUP
70 set Day=%Date:~0,2%
71 set Mth=%Date:~3,2%
72 set Yr=%Date:~6,4%
73 set Hour=%Time:~0,2%
74 if "%hour:~0,1%" == " " set hour=0%hour:~1,1%
75 set Min=%Time:~3,2%
76 set name="%date%-%hour%.%min%"
77
78 cd..
79 mkdir "backup\%name%"
80 xcopy Projekt backup\%name% /E /I /H /Y
81
82 cls
83 echo      ++++++
84 echo      BACKUP
85 echo      ++++++
86 echo.
87 echo.
88 echo      Backup zostal wykonany z data %name%
89 echo.
90 echo.
91 echo      ++++++
92 echo.
93 set /p wybor="Aby wrocic do MENU nacisnij ENTER..."
94 goto MENU

```

---

- liczby\_zaprzyjzaznione.py

```

1      import sys,os
2
3  def sumDivisors(number):
4      newSum = 0
5      for x in range(1, number):
6          if number%x == 0:
7              newSum+=x
8          else:

```

```

9         continue
10    return newSum
11
12 def areFriends(num1, num2):
13     k=""
14     if num1!=num2:
15         a=sumDivisors(num1)
16         b=sumDivisors(num2)
17         if num1==b and num2==a:
18             k="Liczby "+str(num1)+" oraz "+ str(num2)+" sa razem
                zaprzyjaznione! :)"
19         else:
20             k="Liczby "+str(num1)+" oraz "+str(num2)+" nie sa
                zaprzyjaznione! :("
21     else:
22         k = "TO TA SAMA LICZBA! Oszustwo!"
23
24     return k
25
26 def LoadFile(fol):
27     dirname = os.path.dirname(__file__)
28     filename = os.path.join(dirname, fol)
29     text=""
30     try:
31         folder=open(filename, "r")
32         text = folder.readlines()
33     except IOError:
34         print("Blad przy probie otworzenia pliku!")
35         print(filename)
36         exit()
37     return text
38
39 def SaveFile(czyzap, fileName):
40     fil = open(fileName+"_wyniki.txt", "a")
41     fil.write(str(czyzap))
42     fil.write("\nPochodza z pliku: "+fileName+"\n")
43     fil.close()
44
45 def statistics(fileName):
46     yes = 0
47     no = 0
48     lie = 0
49     newName=fileName+"_wyniki.txt"
50     fol = LoadFile(newName)
51     fol1=[]
52     for n in range(0, len(fol)):
53         fol1+=fol[n].split(" ")
54
55     #fol1=fol.split("\w ")
56     for x in range(0, len(fol1)):
57         if fol1[x]=="nie":
58             no+=1
59         if fol1[x]=="razem":
60             yes+=1
61         if fol1[x]=="SAMA":

```

```

62         lie+=1
63     else:
64         continue
65
66     naTak = "Ilosc par zaprzyjaznionych: "+str(yes)
67     naNie = "Ilosc par niezaprzyjaznionych: "+str(no)
68     together = "Ilosc par przeszukanych: "+str((yes+no))
69     oszustwa = "Ilosc par, ktore byly tymi samymi liczbami: "+str(
        lie)
70     saveName = fileName+"_statystyka.txt"
71     k = open(saveName, "w")
72     k.write(naTak+"\n"+naNie+"\n"+together+"\n"+oszustwa)
73
74
75 l = input("Podaj nazwe pliku bez jego rozszerzenia: ")
76 test = LoadFile(l+".txt")
77 for x in range(0,len(test)):
78     luv = test[x].split(" ")
79     c = int(luv[0])
80     d = int(luv[1])
81     kap=areFriends(c,d)
82     SaveFile(kap, l)
83 statistics(l)

```

---

- LZ\_tworzenieistrony.py

```

1         import webbrowser
2
3 l = input("Podaj poczatek nazwy pliku az do _wynik.txt: ")
4 z = l+"_wyniki.txt"
5 n = l+"_statystyka.txt"
6 name=l+"_zaprzyjaznione.html"
7 f = open(name, 'w')
8 table=" "
9 i=0
10
11 #Generowanie tabeli
12 try:
13     plik = open(z)
14     for line in plik:
15         if(i%2==0):
16             table += "<tr>"
17             table+="<td>"+line
18             i+=1
19 except IOError:
20     print("Blad otwarcia pliku!")
21     exit()
22
23 #Wczytywanie statystyk
24 try:
25     stat = open(n)
26     statystyka="<h4>"
27     for line in stat:
28         statystyka+=line+"<br>"

```



```

29     statystyka+="</h4>"
30 except IOError:
31     print("Bład otwarcia pliku!")
32     exit()
33
34 #Budowa strony
35 ttop=""
36 <table>
37     <thead>
38         <tr> <th> Czy liczby sa zaprzyjaznione? <th> Skad?
39     <tbody>
40 ""
41
42 top = ""
43 <html>
44 <center>
45 <style>
46 body {background-color: CC99FF;}
47 h1 {padding: 30px;padding-bottom: 40px;}
48 th, td, table {border: 1px solid black}
49 th{width: 33%;}
50 table{background-color:#CC66FF; margin-bottom:15px;}
51 thead tr, thead th{border: 3px solid black; font-size: 20px;
    background-color:#9900CC; withdh: 33%;}
52 td {border-left: 3px solid black;}
53 table {width: 70%; border-collapse: collapse; text-align: center;
    border: 3px solid black}
54 h3 {text-align:center; margin-top: 50px; font-size: 25px}
55 h4 {text-align:center; font-weight: normal; font-size: 25px}
56 </style>
57 <head><h1>PROGRAM LICZB ZAPRZYJAZNIONYCH - WYNIKI </h1></head>
58 <body>
59 ""
60
61 tend = ""
62 </table>
63 ""
64
65 bottom = ""
66 <h3>This website was brought to you by Natalia Szalas.</h3.
67 </body>
68 </center>
69 </html>
70 ""
71
72 #Generowanie calosci
73 f.write(top+ttop+table+tend+statystyka+bottom)
74 f.close()

```

---