

Języki Skrypowe

dokumentacja projektu Antysymetria

Politechnika Śląska
Matematyka Stosowana, Informatyka
Rok II, Semestr III, 2022/2023
Grupa 1/2

Kateryna Dryzhakova

20 stycznia 2023

Część I

Opis programu

Bajtazar studiuje różne napisy złożone z zer i jedynek. Niech będzie takim napisem, przez $I(R)$ będziemy oznaczać odwrócony (czyli "czytany wspak") napis I , a przez U będziemy oznaczać napis powstały z I przez zamianę wszystkich zer na jedynki, a jedynek na zera.

Bajtazara interesuje antysymetria, natomiast niezbyt lubi wszystko co symetryczne. Antysymetria nie jest tylko prostym zaprzeczeniem symetrii. Powiemy, że (niepusty) napis I jest antysymetryczny, jeżeli dla każdej pozycji i w I , i -ty znak od końca jest różny od i -tego znaku, licząc od początku. W szczególności, niepusty napis złożony z zer i jedynek jest antysymetryczny wtedy i tylko wtedy, gdy $I = U(R)$. Na przykład, napisy 00001111 i 010101 są antysymetryczne, natomiast 1001 nie jest.

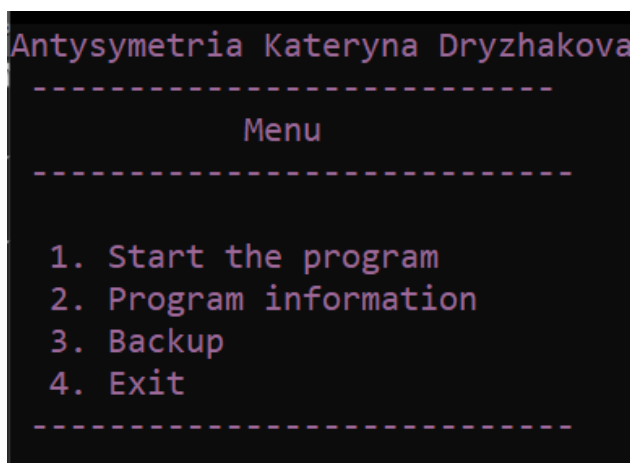
W zadanym napisie złożonym z zer i jedynek chcielibyśmy wyznaczyć liczbę jego spójnych (tj. jednokawałkowych) niepustych fragmentów, które są antysymetryczne. Jeżeli różne fragmenty odpowiadają takim samym słowom, to i tak należy je policzyć wielokrotnie.

Wejście Pierwszy wiersz standardowego wejścia zawiera liczbę n , oznaczającą długość napisu. Drugi wiersz zawiera napis złożony z liter 0 i/lub 1 o długości n . Napis ten nie zawiera żadnych odstępów.

Wyjście Pierwszy i jedyny wiersz standardowego wyjścia powinien zawierać jedną liczbę całkowitą, oznaczającą liczbę spójnych fragmentów wczytanego napisu, które są antysymetryczne.

Instrukcja obsługi

Aby uruchomić program należy włączyć skrypt menu.bat otwierający menu obsługi naszego programu. Po uruchomieniu wyświetli nam się tekst z instrukcją obsługi programu, wymagający podania przez użytkownika liczby w celu wykonania odpowiadającej mu funkcji.



Rysunek 1: Główne menu programu

Możliwe wybory są następujące:

1. Start the program - Uruchamia program pobierając przy tym wszystkie dane przez użytkownika i tworzy raport.html.

```
-----
Menu
-----

1. Start the program
2. Program information
3. Backup
4. Exit
-----

Your choice(1-4) 1

Enter your number > 11001011
Amount of antisymmetrics fragments > 7
Antisymmetric fragments > 1 1 0 0 1 0 0 0 1 0 1 1 0 1 0 1 0 1 1 0 0 1
```

Rysunek 2: Przykładowy komunikat o pomyślnej próbie uruchomienia programu

Raport 08-01-2023 20:47:24

input	output
11001011	7

Rysunek 3: Przykładowy raport programu

2. Program information - Wypisuje na ekranie konsoli opis założeń programu

```
Your choice(1-4) 2

Текущая кодовая страница: 1251
Bajtazar studiujе różne napisy złożone z zer i jedynek.
Niech I będzie takim napisem, przez I~R będziemy oznaczać odwrocony (czyli "czytany wspak") napis I, a przez I_ będziemy oznaczać napis powstały z I przez zamianę wszystkich zer na jedynki, a jedynek na zera.
Bajtazara interesuje antysymetria, natomiast niezbyt lubi wszystko co symetryczne. Antysymetria nie jest tylko prostym zaprzeczeniem symetrii.
Powieśmy, że (niepusty) napis I jest antysymetryczny, jeżeli dla każdej pozycji i w I, i-ty znak od końca jest różny od i-tego znaku, licząc od początku.
W szczególności, niepusty napis I złożony z zer i jedynek jest antysymetryczny wtedy i tylko wtedy, gdy I ~ I~R. Na przykład, napisy 00001111 i 010101 są antysymetryczne, natomiast 1001 nie jest.
W zadanym napisie złożonym z zer i jedynek chcielibyśmy wyznaczyć liczbę jego spójnych (tj. jednokawałkowych) niepustych fragmentów, które są antysymetryczne. Jeżeli różne fragmenty odpowiadają takim samym słowom, to tak należy je policzyć wielokrotnie.

Wzjście
Pierwszy wiersz standardowego wejścia zawiera liczbę n, oznaczającą długość napisu. Drugi wiersz zawiera napis złożony z liter 0 i/lub 1 o długości n. Napis ten nie zawiera żadnych odstępów.

Wyjście
Pierwszy i jedyny wiersz standardowego wyjścia powinien zawierać jedną liczbę całkowitą, oznaczającą liczbę spójnych fragmentów wczytanego napisu, które są antysymetryczne.

Autor: Kateryna Dnyzhakova grupa 1/2
```

Rysunek 4: Przykładowy raport programu

3. Backup - Tworzy kopię zapasową danych w katalogu backups zawierającą raport.html oraz zawartość folderów input i output

4. Zakoncz - Zamyka menu, kończąc tym samym program.

Podanie innej liczby lub znaku, skutkuje powiadomieniem o wprowadzeniu niepoprawnego polecenia

Your choice(1-4) 3

ROBOCOPY :: Robust File Copy for Windows

Начало: 8 января 2023 г. 21:40:21

Источник : D:\r-Ėŷŕa\Politechnika\Języki skryptowe\Projekt\Moj projekt\input\

Назначение : D:\r-Ėŷŕa\Politechnika\Języki skryptowe\Projekt\Moj projekt\backups\08.01.2023--1-40-21\input\

Файлы: *.*

Параметры: *.* /DCOPY:DA /COPY:DAT /R:1000000 /W:30

100% Новая папка 1 D:\r-Ėŷŕa\Politechnika\Języki skryptowe\Projekt\Moj projekt\input\
Новый файл 8 0.txt

	Всего	Скопировано	Пропущено	Несоответствие	СБОЙ	Дополнения
Каталогов :	1	1	0	0	0	0
Файлов :	1	1	0	0	0	0
Байт :	8	8	0	0	0	0
Время :	0:00:00	0:00:00			0:00:00	0:00:00

Скорость : 8000 Байт/сек.

Скорость : 0.457 МБ/мин.

Окончание: 8 января 2023 г. 21:40:21

ROBOCOPY :: Robust File Copy for Windows

Начало: 8 января 2023 г. 21:40:21

Источник : D:\r-Ėŷŕa\Politechnika\Języki skryptowe\Projekt\Moj projekt\output\

Назначение : D:\r-Ėŷŕa\Politechnika\Języki skryptowe\Projekt\Moj projekt\backups\08.01.2023--1-40-21\output\

Файлы: *.*

Параметры: *.* /DCOPY:DA /COPY:DAT /R:1000000 /W:30

100% Новая папка 1 D:\r-Ėŷŕa\Politechnika\Języki skryptowe\Projekt\Moj projekt\output\
Новый файл 1 0.txt

	Всего	Скопировано	Пропущено	Несоответствие	СБОЙ	Дополнения
Каталогов :	1	1	0	0	0	0
Файлов :	1	1	0	0	0	0
Байт :	1	1	0	0	0	0
Время :	0:00:00	0:00:00			0:00:00	0:00:00

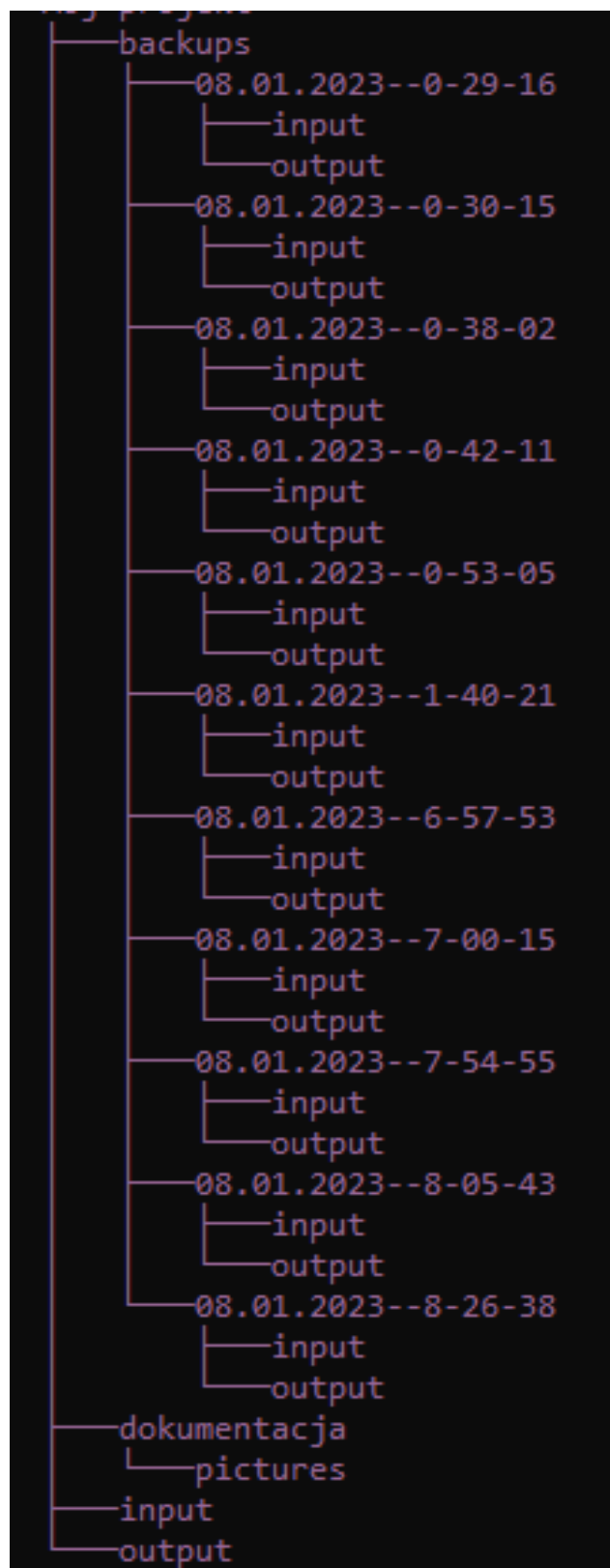
Скорость : 1000 Байт/сек.

Скорость : 0.057 МБ/мин.

Окончание: 8 января 2023 г. 21:40:21

Скопировано файлов: 1.

Rysunek 5: Przykładowy raport programu



Rysunek 6: Przykładowy raport programu

Struktura danych programu:

Program składa się z następującej struktury danych, wymaganych do prawidłowego uruchomienia aplikacji:

- menu.bat - Skrypt batch będący menu, którym uruchamia się program, wyświetla informacje o programie jak i tworzy kopie zapasową danych otrzymanych w wyniku wykonania tegoż programu.

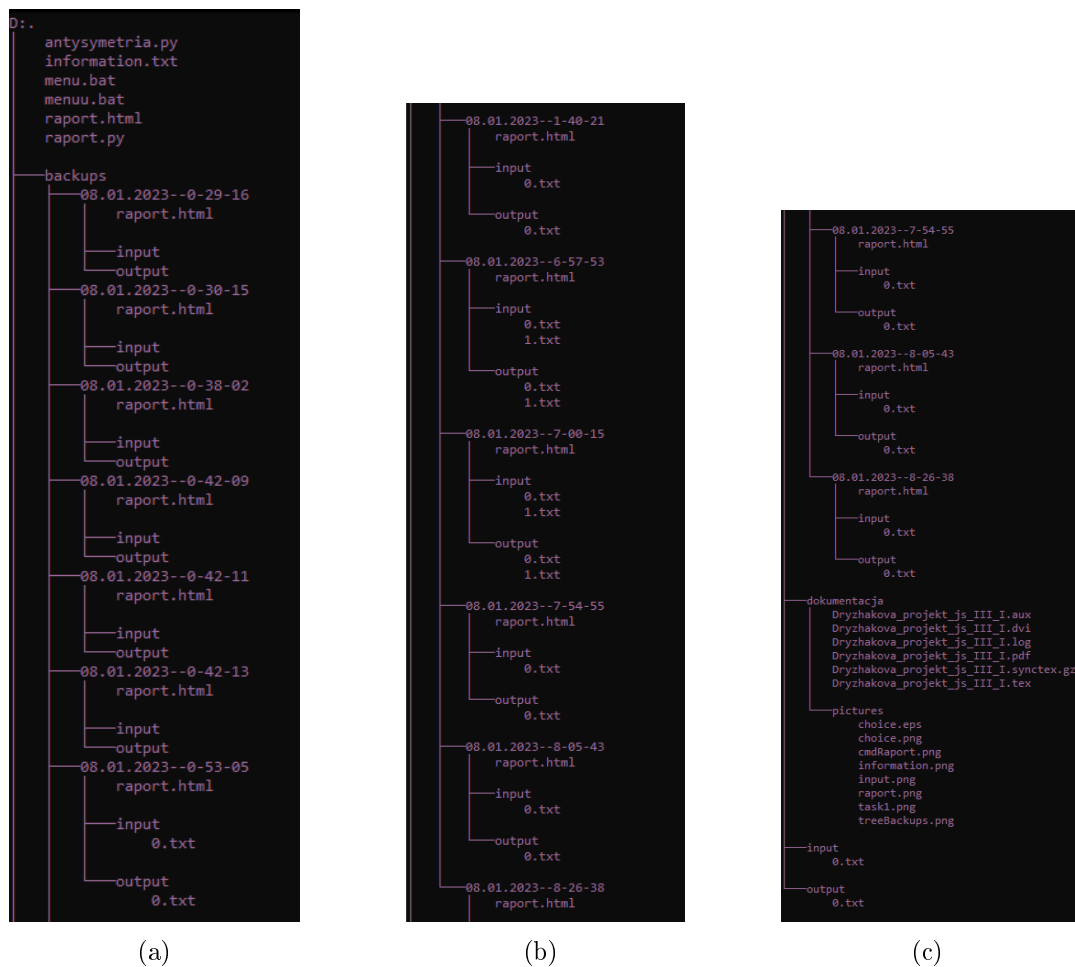
- antysymetria.py - Skrypt python zawierający główny program, pobierający pliki wejściowe zawierające wpisane przez użytkownika liczby i tworzący plik wyjścia zawierający liczbę fragmentów antysymetrycznych.

- raport.py - Skrypt python pobierający dane z plików wejścia oraz wyjścia i generujący plik raport.html zawierający raport wszystkich danych w postaci tabeli.

- Katalog input - pliki wejściowe zawierające zapisane dane przez użytkownika, nazwane według klucza:

$$input0.txt \quad (1)$$

Ponadto program w wyniku działania tworzy dodatkowo katalogi output, backups oraz plik raport.html



Rysunek 7: Struktura danych programu w formie drzewa

Część II

Opis działania

Skrypt `menu.sh` pobiera liczby przez użytkownika i przekazuje pojedynczo jako argument do `antysymetria.py`. Następnie dla pobranej liczby program utworzy pętlę, która na początku sprawdzić czy podana długość zbiega się z długości liczby.

Następnie, przejdzie przez każdy element napisu i sprawdzi, czy wybrany element oraz jego odpowiednik (element, który jest równy liczbie elementów napisu minus indeks elementu plus jeden) są różne. Jeśli tak, to należy zwiększyć licznik antysymetrycznych fragmentów o jeden.

Dalej należy znaleźć wszystkie spójne fragmenty napisu, które są antysymetryczne. Można to zrobić, tworząc pętlę, która przejdzie przez każdy element napisu i sprawdzi, czy jego odpowiednik jest różny od niego. Jeśli tak, to sprawdzamy, czy fragment napisu od obecnego elementu do jego odpowiednika jest antysymetryczny. Jeśli tak, to zwiększamy licznik antysymetrycznych fragmentów o jeden.

Finalnie program kolejno wypisuje zawartość znalezionych fragmentów antysymetrycznych do pliku `output`. Otrzymane w ten sposób wyniki są następnie przetwarzane przez `raport.py`, który tworzy plik `raport.html`, który w tabeli umieszcza zarówno zawartość plików `input` oraz `output`, tylko i wyłącznie jeśli numer w ich nazwie jest taki sam, co sprawia, że raport zawiera jedynie pomyślnie wykonane iteracje programu. Ostatecznie uruchamiana jest domyślna przeglądarka użytkownika, w której wyświetla się utworzony `raport.html`.

Algorytmy

Data: Dane wejściowe pobrane przez użytkownika *dataInput*, *length*

Result: Dane wyjściowe plik output

while *dataInput*! = *length* **do**

 | Enter another correct number;

end

amountAntisymmetricsFragments(*dataInput*, *output*)

Function *amountAntisymmetricsFragments*(*dataInput*, *output*)

for *i* < *dataInput.len* **do**

for *i* + 1 < *dataInput.len* + 1 **do**

if *antisymmetrics*(*dataInput*[*i* : *j*]) **then**

 | Dodaj do output; Obliczyć liczbę wchodzenij;

end

end

end

return *liczbaWchodzenij*;

Function *antisymmetrics*(*dataInput*)

for *i* < *dataInput.len* **do**

if *dataInput*[*i*] < *dataInput*[*-i* - 1] **then**

 | **return** *False*;

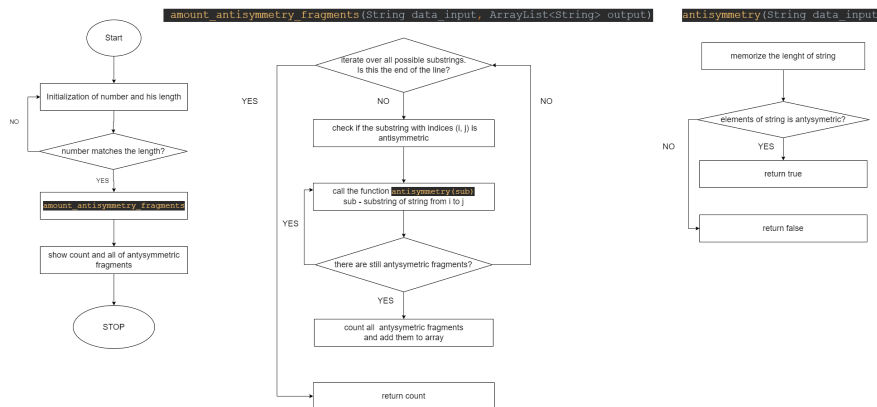
end

return *True*;

end

Przypisz do *output* zawartość function *amountAntisymmetricsFragments*

Algorithm 1: Algorytm wyszukiwania antysymetrii liczby



Rysunek 8: Schemat blokowy

Implementacja systemu

Uruchomienie programu z poziomu skryptu menu.bat powoduje sprawdzenie czy istnieje katalog na pliki wynikowe "output", jeśli tak to usuwa go wraz z zawartością oraz tworzy go na nowo, następnie uruchamia antysymmetria.py wraz z argumentem zadany przez użytkownika.

Jeśli program znalazł wynik i zakończył się powodzeniem, tworzy on plik wynikowy do katalogu "output". Sprawdzane jest czy w katalogu "output" znajdują się jakiekolwiek pliki. Jeśli tak to uruchamiany jest skrypt raport.py który tworzy plik raport.html w głównym katalogu projektu, a jeśli nie to program kończy się z odpowiednim komunikatem.

Opcja wyświetlająca informacje wypisuje dane z pliku "information.txt" znajdującego się w katalogu "mój projekt".

Wybranie funkcji backup sprawdza czy istnieje w głównym katalogu plik raport.html, jeśli tak to tworzy katalog "backups" o ile takowy już nie istnieje, następnie tworzy katalog nazwany aktualną datą i godziną wewnątrz folderu "backups", który natomiast zawiera kopię całego stworzonego katalogu "input", katalogu "output" oraz pliku raport.html.

Wykorzystane biblioteki i przykłady ich użycia

os

```
1     for f in os.listdir(dir):
2         os.remove(os.path.join(dir, f))
3         //sprawdza czy plik istnieje oraz usuwa go
```

```
1     os.chdir("input")
2     //change the current directory to input directory
```

sys

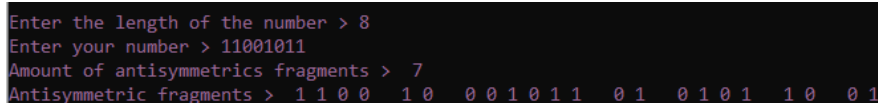
```
1     with open(sys.argv[1], "a+") as g:
2         g.write(str(data_input))
3         //zapisuje podanu liczbę do pliku
```

Testy

Dane wejściowe: Długość 8 Liczba 11001011

Wypisujemy antysymetryczne fragmenty: 01 (pojawia się dwukrotnie), 10 (także dwukrotnie), 0101, 1100 oraz 001011.

Dane wyjściowe: 7



```
Enter the length of the number > 8
Enter your number > 11001011
Amount of antisymmetrics fragments > 7
Antisymmetric fragments > 1 1 0 0 1 0 0 0 1 0 1 1 0 1 0 1 1 0 0 1
```

Rysunek 9: Wynik programu w konsoli

Dane wejściowe: Długość 3 Liczba 11001011

Dane wyjściowe: Powtórz wpisanie liczby

Długość 4 Liczba 1100

Wypisujemy antysymetryczne fragmenty: 1100, 10.

Dane wyjściowe: 2

```

Enter the length of the number > 3
Enter your number > 11001011
The string you entered is not the correct length. Please, try again > 101
Amount of antisymmetrics fragments > 2
Antisymmetric fragments > 1 0 0 1

```

Rysunek 10: Wynik programu w konsoli

```

Enter the length of the number > 4
Enter your number > 1100
Amount of antisymmetrics fragments > 2
Antisymmetric fragments > 1 1 0 0 1 0

```

Rysunek 11: Wynik programu w konsoli

Pełen kod aplikacji

menu.bat

```

1 @echo off
2
3 rem set choice_from_cli = %1
4 echo Antysymetria Kateryna Dryzhakova
5 :menu
6 echo -----
7 echo                      Menu
8 echo -----
9 echo .
10 echo 1. Start the program
11 echo 2. Program information
12 echo 3. Backup
13 echo 4. Exit
14 echo -----
15 echo .
16 set /p choice=Your choice (1-4)
17
18 if %choice%==1 goto task1
19 if %choice%==2 goto task2
20 if %choice%==3 goto task3
21 if %choice%==4 goto exit
22 echo Your choice is not included in the range 1-4
23 goto menu
24 :task1
25 echo .
26 IF EXIST raport.html DEL raport.html
27 IF NOT EXIST output mkdir output
28 echo "<HTML>" >> raport.html
29 DEL /Q output
30 for /f "delims=" %%a in ('dir /b input') do (
31     call python antysymetria.py %%a
32 )
33 call python raport.py
34 echo .

```

```

35 goto menu
36 :task2
37 echo.
38 chcp 1251
39 type information.txt
40 pause
41 echo.
42 goto menu
43 :task3
44 echo.
45 IF NOT EXIST backups mkdir backups
46 set name=%date%--%TIME:~1,7%
47 set name=%name:=-%
48 IF EXIST raport.html mkdir backups\%name%
49 robocopy input backups\%name%\input
50 robocopy output backups\%name%\output
51 copy raport.html backups\%name%\raport.html
52 echo.
53 goto menu
54 :exit

```

antysymetria.py

```

1 import sys
2 import os
3
4 def antisymmetrics(data_input):
5     for i in range(len(data_input)):
6         if data_input[i] == data_input[-i-1]:
7             return False
8     return True
9
10 def amount_antisymmetrics_fragments(data_input, output):
11     count = 0
12     for i in range(len(data_input)):
13         for j in range(i+1, len(data_input)+1):
14             # sprawdzamy, czy podnapis o indeksach (i, j) jest
15             # antysymetryczny
16             if antisymmetrics(data_input[i:j]):
17                 output += data_input[i:j]
18                 output += " "
19                 count += 1
20     return count
21
22 dir = D:\university\Politechnika\Języki skryptowe\Projekt\Moj projekt\
23     input
24
25 for f in os.listdir(dir):
26     os.remove(os.path.join(dir, f))
27
28 length = int(input("Enter the length of the number > "))
29 data_input = input("Enter your number > ")
30
31 while(len(data_input) != length):
32     data_input = input("The string you entered is not the correct length.
33     Please, try again > ")

```

```

30
31 os.chdir("input")
32
33 with open(sys.argv[1], "a+") as g:
34     g.write(str(data_input))
35
36 output = []
37
38 amount = amount_antisymmetrics_fragments(data_input, output)
39 print("Amount of antisymmetrics fragments > ", amount)
40 print("Antisymmetric fragments > ", *output)
41
42 os.chdir("..")
43 os.chdir("output")
44 with open(sys.argv[1], "a+") as g:
45     g.write(str(amount))

```

raport.py

```

1 import os
2 from datetime import date, datetime
3 from os.path import isfile, join, exists
4
5 now = datetime.now()
6 fulldate = now.strftime("%d-%m-%Y %H:%M:%S")
7
8 outputfile = open("Raport.html", "w")
9
10 outputfile.write(f"""
11 <html>
12     <head>
13         <title>Raport Antysymetria</title>
14     </head>
15     <body>
16         <h1>Raport {fulldate}</h1>
17         <table>
18             <tr>
19                 <th>input</th>
20                 <th>output</th>
21             <tr>
22                 """
23
24
25 inputfiles = [file for file in os.listdir("input") if isfile(join("input", file))]
26 for x in range(len(inputfiles)):
27     with open(f"input/{x}.txt", "r") as f:
28         lines = [str(line.rstrip()) for line in f]
29         outputfile.write(f""" <tr><td>""" + str(lines).replace("'", " ")
30                             [1:-1])
31
32         outputfile.write(f"""</td>
33                             <td>"""
34
35         with open(f"output/{x}.txt", "r") as f:
36             lines = [str(line.rstrip()) for line in f]

```

```
35     outputfile.write(str(lines).replace("'", " ")[1:-1])
36     outputfile.write("""</td>
37         </tr>""")
38 outputfile.write("""
39     </table>
40 </body>
41 </html>""")
42
43 outputfile.close()
```
