File Handling

# Before Class

1. Zapoznaj się z materiałem zawartym w lekcji 29 dostępnej na kanale „Introduction to Computer Science and Programming Using Python (MIT)”.

<https://www.youtube.com/playlist?list=PLRJdqdXieSHN0U9AdnmwD-9QcR9hmw04d>

1. Korzystając z Python Tutorial zapoznaj się z wprowadzeniem do języka Python – punkty 7.2 oraz 7.2.1. Spróbuj wykonać przedstawione w tym punkcie polecenia na swoim komputerze (w trybie interaktywnym, w aplikacji Thonny).

<https://docs.python.org/3/tutorial/index.html>

1. Na platformie w3schools.com wykonaj lekcje dostępne w Python Tutorial – sekcja File Handling oraz PythonRegEx. Spróbuj również wykonać zawarte w tutorialu przykłady na swoim komputerze.

<https://www.w3schools.com/python/default.asp>

1. Zapoznaj się z materiałem dotyczącym wyrażeń regularnych
   1. Wikipedia: <https://pl.wikipedia.org/wiki/Wyra%C5%BCenie_regularne>
   2. Środowisko programisty/Wyrażenia regularne: <http://wazniak.mimuw.edu.pl/index.php?title=%C5%9Arodowisko_programisty/Wyra%C5%BCenia_regularne>
   3. Wprowadzenie do wyrażeń regularnych:  
      <https://czterytygodnie.pl/wprowadzenie-wyrazen-regularnych-regex/>
   4. Regex tutorial: <https://medium.com/factory-mind/regex-tutorial-a-simple-cheatsheet-by-examples-649dc1c3f285>
   5. Regular Expression HOWTO: <https://docs.python.org/3/howto/regex.html>
   6. Regular expressions online: <https://regex101.com/>

# During Class

## Reading from file

1. Pobierz z dysku lub utwórz w edytorze znakowym plik tekstowy NoEducation.txt zawierający poniższy fragment utworu angielskiego zespołu rockowego Pink Floyd:

Pink Floyd – We Don't Need No Education  
=======================================  
We don't need no education   
We don't need no thought control   
No dark sarcasm in the classroom   
Teacher, leave them kids alone   
Hey! Teacher, leave them kids alone

Następnie napisz program, który wyświetli utworzony plik tekstowy.

# wariant 1  
file = open('C:/Users/Janusz/NoEducation.txt','r')  
print( file.read() )  
file.close()

# wariant 2  
with open('C:/Users/Janusz/NoEducation.txt','r') as file:  
 print( file.read() )

1. Napisz program wyświetlający zawartość pliku tekstowego NoEducation.txt. Na początku każdej linii wyświetl numer kolejnego wiersza. Wykorzystaj poniższy fragment programu.

with open('C:/Users/Janusz/NoEducation.txt','r') as file:  
 for line in file:  
 print(line, end='')

1. Plik numbers.txt zawiera wykaz liczb naturalnych. Napisz program obliczający sumę tych liczb. Wskazówka: odczytaj kolejny wiersz z pliku i dokonaj jego konwersji do wartości numerycznej. Porównaj rezultat z wynikami uzyskanymi przez innych studentów.

## Writing to file

1. Napisz program, który utworzy plik PersonalData.txt. Zapisz do pliku swoje imię i nazwisko, nazwę uczelni oraz kierunek studiów (w trzech oddzielnych liniach). Sprawdź w dowolnym edytorze znakowym, czy plik zawiera właściwe dane. Wskazówka: otwórz plik w trybie do zapisu i wykorzystaj metodę write().

with open('DanePersonalne.txt','w') as file:  
 file.write('Jan Nowak\n')  
 …  
 …

1. Napisz program, który umożliwi dodanie do pliku tekstowego shoppinglist.txt nazwy kolejnego produktu, który należy zakupić. Nazwę produktu wprowadzaj z klawiatury. Wskazówka: otwórz plik w trybie do dodawania.
2. Tablica zawiera następujące wartości całkowite: 32, 16, 5, 8, 24, 7. Napisz program zapisujący zawartość tablicy do pliku tekstowego, każda liczba w oddzielnym wierszu.

## Regular expressions

1. Skopiuj zawartość pliku land.txt na stronę internetową <https://regex101.com>. Następnie utwórz wyrażenia regularne, które wskażą w tekście:
   1. wszystkie znaki
   2. słowa ‘Poland’
   3. nazwy państw (Poland, Germany oraz France)
   4. litery
   5. wielkie litery
   6. cyfry
   7. wszystkie znaki oprócz cyfr
   8. odstępy
   9. samogłoski
   10. liczby reprezentujące rok
   11. liczby wyrażone procentowo
   12. znaki interpunkcyjne
   13. słowa
   14. słowa rozpoczynające się literą ‘n’
   15. słowa trzyliterowe
   16. słowa składające się z co najmniej pięciu liter
   17. słowa rozpoczynające się wielką literą
   18. słowa ‘the’
   19. spójniki ‘and’
   20. zdania
   21. fragmenty tekstu ujęte w nawiasy
2. Poniższy informacje zawierają dane dotyczące uruchomionych kursów (kod kursu, pełna nazwa kursu, liczba uczestników).

MAT Matematyka 45  
GEO Geografia 210  
HIS Historia 28  
INF Informatyka 196  
AST Astronomia 9

Korzystając z serwisu https://regex101.com utwórz wyrażenia regularne, które wskażą:

* 1. Kody przedmiotów
  2. Nazwy przedmiotów
  3. Liczby uczestników

1. Komunikat dotyczący prognozowanej temperatury w stopniach Celsjusza na kolejne trzy dni został opublikowany w Internecie:

'wtorek - 23C, środa - 21C, czwartek 25C '

Napisz program, który obliczy średnią prognozowaną temperaturę. Wykorzystaj wyrażenia regularne do wyodrębnienia z komunikatu prognozowanej temperatury w kolejnych dniach.

import re  
  
komunikat = 'wtorek - 23C, środa - 17C, czwartek 25C'  
cyfry = re.findall('\d{2}',komunikat)  
# uzupełnij program  
# ...

1. Napisz program, który obliczy liczbę samogłosek w tekście:

To be, or not to be, that is the question

Wykorzystaj wyrażenia regularne oraz metodę findall().

# After Class

1. Plik numbers.txt zawiera wykaz liczb naturalnych. Napisz program wyświetlający te liczby jako ciąg rosnący. Wskazówka: dodaj odczytane liczby z pliku do tablicy (wykorzystaj metodę append() ), a następnie uporządkuj tablicę rosnąco wykorzystując metodę sort(). Wyświetl uporządkowaną tablicę liczb rozdzielając je znakiem odstępu.
2. Plik universities.txt zawiera nazwy uniwersytetów. Napisz program, który uporządkuje je alfabetycznie. Wskazówka: odczytaj wszystkie nazwy do tablicy i uporządkuj je alfabetycznie. Następnie zapisz zawartość tablicy do pliku, nadpisując jego poprzednią zawartość.
3. Napisz program, który zapisze do pliku evennumbers.txt wszystkie naturalne liczby parzyste znajdujące się w pliku numbers.txt.
4. Plik numbersinrows.txt zawiera wykaz liczb naturalnych (kilka w każdym wierszu). Napisz program obliczający, ile liczb znajduje się w pliku oraz jaka jest ich suma. Wskazówka: wykorzystaj metodę split(), aby wyodrębnić liczby w każdym wierszu. Porównaj rezultat z wynikami uzyskanymi przez innych studentów.
5. W edytorze znakowym utwórz plik tekstowy o nazwie students.txt zawierający poniższe dane:

first\_name,last\_name,age,gender,email  
Decca,Blackstone,52,Male,dblackstone0@time.com  
Elena,Rechert,27,Female,erechert1@ucoz.com  
Bibbye,Norree,26,Female,bnorree2@reddit.com  
Alasdair,McCoole,53,Male,amccoole3@foxnews.com  
Hogan,Hatrey,26,Male,hhatrey4@zimbio.com

Napisz program, który odczyta dane z pliku tekstowego i wyświetli imię, nazwisko oraz email osób mających mniej, niż 30 lat. Sformatuj dane, jak poniżej. Przykładowy rezultat:

Elena Rechert erechert1@ucoz.com  
Bibbye Norree bnorree2@reddit.com  
Hogan Hatrey hhatrey4@zimbio.com

1. Napisz program, który obliczy sumę cyfr występującą w pliku land.txt. Wykorzystaj wyrażenia regularne.
2. Tablica dwuwymiarowa zawiera poniższe wartości:

[['Marek','Zelnik','zelnik@sed.pl'],['Ewa','Maj','maje@wp.pl'],['Piotr','Wyga','wygap@gop.pl']]

Napisz program, który dane zawarte w tablicy zapisze do pliku tekstowego w formacie CSV (https://en.wikipedia.org/wiki/Comma-separated\_values). Następnie spróbuj otworzyć ten plik w dowolnym arkuszu kalkulacyjnym. Przykładowy rezultat:

Imie,Nazwisko,Email  
Marek,Zelnik,zelnik@sed.pl  
Ewa,Maj,maje@wp.pl  
Piotr,Wyga,wygap@gop.pl