**I.1) Co to jest lista, krotka, słownik i zbiór?**

Lista – czyli tablica, zmienne ustawione w szereg  
Krotka – tuple, podobnie jak tablica, ale ich zawartość jest zasadniczo stała (krotka może zawierać listę której elementy możemy edytować)  
Słownik – podobnie jak tablica, ale zamiast indeksów wykorzystuje klucze  
Zbiór – tablica w której elementy się nie powtarzają

**I.2) Jak się definiuje i wykorzystuje te obiekty?**  
Lista:  
 lista = [] #dodajemy elementy przez .append()  
 lista = [1,2,3]  
Krotka:  
 dane = ("Aaa", 1, 2.13) #dane możemy tylko odczytać  
Słownik:  
 kontakty = {}  
 kontakty["Jan"] = 938477566 #odczytując klucz ‘Jan’ dostaniemy nr tel  
Zbiór:  
 A = set(["aaa", "bbb", "ccc"])

**I.3) Wyjątki i ich obsługa.**  
 try:

(polecenie)

except (nazwaWyjątku):

(polecenie gdy wystąpi wyjątek)  
  
**I.4) Operacje na plikach. Różnice między obsługą plików tekstowych, a binarnych.**

Zapisując plik binarnie musimy odpowiednio formatować to co chcemy zapisać, o wiele prościej jest skorzystać z modulu pickle  
 plik = open('ścieżka/do/pliku.txt', ‘(tryb)’)

#tryb – r-odczyt, w-zapis, a-dopis + t-tekstowy (domyślnie), b-binarny  
 plik.write( „coś” ) #tekstowo, tylko gdy zapis i dopis  
 print (plik.read()) #tekstowo, tylko gdy odczyt

plik.close()  
Inny sposób odczytu/zapisu linia po linii:

with open('plik.txt', ‘rt’) as plik: #odczyt linia po linii

for linia in plik:

print linia.strip().split()

**I.5) Serializacja i deserializacja danych przy zapisie do i odczycie z pliku binarnego (moduł pickle).**  
Serializacja: przekształcenie danych w strumień bajtów  
Deserializacja: proces odwrotny, zamiana strumienia bajtów w użyteczne dane  
  
 import pickle

output\_file = open("myfile.bin", "wb") #ustawienie do zapisu binarnego

myint = 42

mystring = "Hello, world!"

pickle.dump(myint, output\_file) #serializacja zapis inta jako bajty

pickle.dump(mystring, output\_file) #pickle.dump() do zapisu

output\_file.close()

input\_file = open("myfile.bin", "rb") #ustawienie do odczytu binarnego  
 myint = pickle.load(input\_file) #odczyt bajtów do zmiennej

mystring = pickle.load(input\_file) #pickle.load() do odczytu  
 input\_file.close()