

Neptun azonosító / Neptun id:

Feladat1

A paraméterben kapott bináris fájlokat olvassuk be és írassuk ki az első record tartalmát a standard outputra! (az unpack visszatérési értékét)

- Parameter1 formátuma: float, bool, karakter
- Parameter2 formátuma: karakter, 9 hosszú string, integer
- Parameter3 formátuma: integer, bool, float
- Parameter4 formátuma: karakter, float, 9 hosszú string

Feladat2

Írd ki a stdout-ra (print) a következő értéket bináris formátumban (a pack visszatérési értéke)! A string hosszát a szöveg mögött lévő szám jelzi!

Használandó struct paraméterek: f, i, c, ?, Xs (ahol a X a string hossza, pl: 3s)

- "elso"(16), 74, True
- 77.5, False, 'X'
- 65, "masodik"(14), 84.9
- 'Z', 96, "harmadik"(17)

Script paraméterezése:

- python3 client.py < file1 > < file2 > < file3 > < file4 >
 - pl: python3 client.py db1.bin db2.bin db3.bin db4.bin

Példa kimenet:

```
(b'F', b'123456789', 35)
(35, 37.29999923706055, True)
(True, b'123456789', b'F')
(35, b'F', 37.29999923706055)
b'elso\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00S\x00\x00\x00\x01'
b'\x00\x00\xadB\x00X'
b'J\x00\x00\x00masodik\x00\x00\x00\x00\x00\x00\x00\x00\x00\xcd\xcc\xbbB'
b'Z\x00\x00\x00i\x00\x00\x00harmadik\x00\x00\x00\x00\x00\x00\x00\x00'
```

Leadás: A program leadása a TMS rendszeren keresztül **.zip** formátumban, amiben egy **client.py** szerepeljen!

#####

Task1

Read the binary files received in the parameter and print the contents of the first record to the standard output! (the return value of the unpack)

- Parameter1 structure: float, bool, char
- Parameter2 structure: char, 9 length string, integer
- Parameter3 structure: integer, bool, float
- Parameter4 structure: char, float, 9 length string

Task2

Write the following value to stdout (print) in binary format (the return value of the pack)! The length of the string is indicated by the number behind the text!

Struct parameter to use: f, i, c, ?, Xs (where X is the length of the string, eg: 3s)

- "elso"(16), 74, True
- 77.5, False, 'X'
- 65, "masodik"(14), 84.9
- 'Z', 96, "harmadik"(17)

Parameters of the script:

- python3 client.py < file1 > < file2 > < file3 > < file4 >
 - pl: python3 client.py db1.bin db2.bin db3.bin db4.bin

Example output:

```
(b'F', b'123456789', 35)
(35, 37.29999923706055, True)
```

```
(True, b'123456789', b'F')  
(35, b'F', 37.29999923706055)  
b'elso\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00S\x00\x00\x00\x01'  
b'\x00\x00\xadB\x00X'  
b'J\x00\x00\x00masodik\x00\x00\x00\x00\x00\x00\x00\x00\xcd\xcc\xbbB'  
b'Z\x00\x00\x00i\x00\x00\x00harmadik\x00\x00\x00\x00\x00\x00\x00'
```

Submission: Send it to the TMS in **.zip** file which should contain one and only one file called **client.py**!