

Programming Language, Quiz E180514

Write 4 Python programs to solve the following questions. Please name your program files as *Q1.py*, *Q2.py*, and so on, *i.e.*, according to the serial number of questions. All data files you need can be obtained from the e3 system. Compress your programs along with the data files into a ZIP file and submit the ZIP to e3 by the end of the exam.

1. There are horses and chickens in a farm. Given that there are 1000 heads and 3330 legs in total, please write a program to compute how many horses and chickens, respectively, there are in the farm. Note that loop(s) MUST be used in this program.
2. Protein Data Bank (PDB) is a database for researchers to deposit (or publish) the structure of proteins they determined. Structure data deposited in PDB are stored in the PDB file format. In a PDB file, information of an atom is described by a line starting with "ATOM". As shown in the following figure, in an ATOM line, the "residue type" and "residue ID" of the atom is saved at positions "18–20" and "23–27", respectively. Because a residue is composed of several atoms, it usually takes several ATOM lines to describe a residue. For instance, in the following figure, there are three residues, serine 24, histidine 25 and methionine 26.

		Residue type (amino acid)		Residue ID							
ATOM	1	N	SER	A	24	-8.951	-8.250	-13.831	1.00	0.00	N
ATOM	2	CA	SER	A	24	-8.966	-8.695	-12.448	1.00	0.00	C
ATOM	3	C	SER	A	24	-8.430	-10.125	-12.352	1.00	0.00	C
ATOM	4	O	SER	A	24	-8.954	-11.032	-12.997	1.00	0.00	O
ATOM	11	N	HIS	A	25	-7.394	-10.281	-11.541	1.00	0.00	N
ATOM	12	CA	HIS	A	25	-6.782	-11.586	-11.353	1.00	0.00	C
ATOM	13	C	HIS	A	25	-7.626	-12.413	-10.381	1.00	0.00	C
ATOM	14	O	HIS	A	25	-7.998	-13.546	-10.685	1.00	0.00	O
ATOM	27	N	MET	A	26	-7.905	-11.815	-9.232	1.00	0.00	N
ATOM	28	CA	MET	A	26	-8.698	-12.481	-8.214	1.00	0.00	C
ATOM	29	C	MET	A	26	-9.591	-11.484	-7.474	1.00	0.00	C
ATOM	30	O	MET	A	26	-9.098	-10.539	-6.860	1.00	0.00	O

There are 10 PDB structure files in the **PDB** folder. Write a program to count the number of residues each file has and print the results to the screen. Your output may look like this (no need to sort the results):

```
0tmpA.pdb:    23 residues
1mplA.pdb:   111 residues
1or4A.pdb:   169 residues
..... (etc.)
```

Hint: To know how many items there are in a dictionary, you may use the function **len()**.

3. File `amino_acid_codes.txt` provides the three-lettered and one-lettered codes of amino acids. Lines starting with “#” in this file are comments describing the file format. Write a program to generate a dictionary mapping three-lettered codes to one-lettered codes and use this dictionary to transform each structure file in the `PDB` folder into a one-lettered code amino acid sequence. Output the sequences you obtained to the screen in the fasta format demonstrated below,

```
>0tmpA
SEKSKLQEIIYQELTQLKAAVGEL
>1mp1A
SHMQLKFAECLKVKVDMKVNLEVIKPVITKRVTEILGFEDDVVIEFIFNQLEVKNPDSKMMQINLTGF
LNGKNAREFMGELWPLLLSAQENIAGIPSAFLELKKEEIKQR
>1or4A
ETAYFSDSNGQQKNRIQLTNKHADVKKQLKMVR..... (etc.)
```

4. Given any numeric list, compute the median of this list using a user-defined function.
5. Compose an user-defined function named `strLen()`, which takes a string argument and return the length of the string. For instance,

```
assert strLen("ABCDE") == 5
assert strLen("13579246810") == 11
assert strLen("") == 0
```

6. A ball falls vertically from a place of 100 meters height and bounces back several times. The height of each bounce is 50% of the previous one. Compute the total vertical distance this ball has travelled before its 11th bounce (that is, at the 10th time the ball reaches the ground).
7. Write a program to make the same output as what shown below,

```
1x1=1
2x1=2    2x2=4
3x1=3    3x2=6    3x3=9
4x1=4    4x2=8    4x3=12    4x4=16
5x1=5    5x2=10    5x3=15    5x4=20    5x5=25
6x1=6    6x2=12    6x3=18    6x4=24    6x5=30    6x6=36
7x1=7    7x2=14    7x3=21    7x4=28    7x5=35    7x6=42    7x7=49
8x1=8    8x2=16    8x3=24    8x4=32    8x5=40    8x6=48    8x7=56    8x8=64
9x1=9    9x2=18    9x3=27    9x4=36    9x5=45    9x6=54    9x7=63    9x8=72    9x9=81
```