```
# -*- coding: utf-8 -*-
"""
# Created on Tue Mar  3 14:57:41 2020
# @author: Kania A., Sarapata K., Gucwa M., and Wójcik-Augustyn A.
# This software is released under the GNU General Public License
# Please cite:

# Kania A., Sarapata K., Gucwa M., and Wójcik-Augustyn A.,
# Optimal solution to torsional coefficients fitting problem in force fields parameterization,
# Journal of Physical Chemistry 2021

Analytical solution of dihedral fitting

The script 'K_fit.py' calculates (fits) Fourier parameters for dihedral energy part (in force field equation).
As a input it takes the following data files. First, the energy of a molecule for different conformations is
demanded. It can be obtained using quantum chemistry methods. The second file should include energies for the
same states but without dihedral part (parameters are set to zero). The last input files consist of dihedral
angles of a molecule which were considered (one dih file for every dihedral angle).

According to the formula in cited paper, the script calculates the estimated value of torsion part based on the
given energy values. The calculated parameters are selected in that way to minimize the square of the difference
between corresponding energies from the first and second file. All torsion angles can be grouped according to
their types. For instance the torsion angle formed from OH.8-P.6-OS.10-CT atoms belongs to the same group as
this one: CT.1-OS.5-P.6-OH.8.
The same set of four parameters K1, K2, K3, K4 applies to every dihedral of a given type.
The 'K_fit.py' script optionally expects two values 'from' and 'to' which set the range of the resulting
parameters K. These optionally values and mandatory paths to data files the script gets from the standard input
or text file with paths to QM energy, MM energy and all dihedral angles files.

In the result, the script returns the set of four optimal Fourier or Ryckaert-Belleman (RB) parameters for each
dihedral types and optionally set of four Fourier parameters from declared range for each dihedral types.
The cited work includes the method of obtaining Ryckaert-Belleman parameters based on Fourier ones.

usage:

./K_fit.py from to < paths2files.txt  > coefficients.out

options:

from, to       optionally two values 'from' and 'to' which set the range of the resulting parameters K.
               The calculated K parameters should be included inside this range. This option is only for Fourier
               parameters.

path2files.txt includes paths to data files:
                              path to QM energy file,
                              MM energy file,
                              *.dih files,
                              type of coefficients: Fourier or RB (Ryckaert-Belleman),
                              and the amount of coefficients to calculate (3 or 4 value is allowed).
data files:

QM energy file the energy of a molecule for different conformations with respect to the most stable conformer
e.g. relative energy values from Gaussian used as a base for making a fit. Values (in kJ/mol - the unit has to
be the same like in MM energy file) should have column layout.

MM energy file the relative energy of a molecule for different conformations (with respect to the most stable)
calculated with K's parameters set to 0. Values (in kJ/mol - the unit has to be the same like in QM energy file)
should have column layout.

*.dih files    with values of all dihedrals in molecule for different conformations. Values should have column
               layout. The first line in each files containing the type of dihedral: order of the atoms forming
               that angle. e.g. 'P.6-OS.10-CT.11-HC.14.dih' file in first line should have: 'P OS CT HC'.
               Ordering of values in all *.dih files is the same, so rows are related across all files (see
               table below).

coffType       return the Ryckaert-Belleman coefficients instead of Fourier ones

coff           amount of Fourier coefficients to calculate. Default is 4, another possible one is 3.

All QM, MM and *.dih files should include data in the same order which corresponds to the mutual relationship.
The path to those files are included in paths2files.txt

output:

In the result, the script returns the set of optimal Fourier or RB parameters for each dihedral types
and optionally set of Fourier/RB parameters from declared range for each dihedral types.
The values are in JSON data interchange format.

Example result of calculating of coefficients for sets in path2file are included in coefficints.out file.
"""
```

	15 x *.dih															1 x *.ene	
geometry	CT.1-OS.5-P.6-O2.7	CT.1-OS.5-P.6-OH.8	CT.1-OS.5-P.6-OS.10	HC.2-CT.1-OS.5-P.6	HC.3-CT.1-OS.5-P.6	HC.4-CT.1-OS.5-P.6	O2.7-P.6-OH.8-HO.9	O2.7-P.6-OS.10-CT.11	OH.8-P.6-OS.10-CT.11	OS.10-P.6-OH.8-HO.9	OS.5-P.6-OH.8-HO.9	OS.5-P.6-OS.10-CT.11	P.6-OS.10-CT.11-HC.12	P.6-OS.10-CT.11-HC.13	P.6-OS.10-CT.11-HC.14	QM	MM $E_{zero\text{-}tors}$
1	128.941	-106.885	1.041	61.095	-60.597	-179.728	5.209	53.654	-71.586	133.211	-120.187	179.960	-57.887	63.826	-177.178	9.0864	6.1307
2	139.300	-95.970	11.068	60.407	-61.322	179.425	14.921	54.486	-70.632	143.110	-110.110	179.949	-57.494	64.180	-176.794	7.8332	5.9428
3	149.415	-85.327	20.984	57.760	-64.011	176.749	22.481	55.378	-69.737	150.737	-102.411	-179.678	-57.389	64.364	-176.597	5.7962	5.6458
4	159.287	-74.956	30.966	56.543	-65.233	175.446	27.271	56.169	-69.206	155.582	-97.552	-179.391	-57.233	64.466	-176.409	3.4043	5.0059
5	168.868	-64.958	41.015	55.209	-66.514	174.244	29.161	57.275	-68.498	157.472	-95.679	-178.836	-57.078	64.692	-176.248	1.2732	3.9259
…	…																
…	…																
540	171.379	…															

Table. Each file contains information about the value of one angle for each of 540 geometries. If we would like to obtain a set of angles for a given geometry, then we would need to take one value from a specific row for each of the 15 * .dih files. ($geometry$: $15\ dihedrals\ x\ 36\ (0, 10, 20, 30, \ldots, 350) = 540$)