#!/usr/bin/env python3

# -\*- coding: utf-8 -\*-

from dataset import Dataset, file2lists

from task6\_module import run, make\_result, print\_report

class Datasource(Dataset):

"""docstring for Datasource"""

def \_\_init\_\_(self, filename = None, min\_current = None, max\_temperature = None):

super().\_\_init\_\_(filename)

if filename is None:

self.control\_list, self.probe\_1\_list, self.probe\_2\_list = [], [], []

self.min\_current = min\_current

self.max\_temperature = max\_temperature

@staticmethod

def \_get(get, cond):

return make\_result(\*get(), condition = cond)

def \_\_add\_\_(self, other):

super().\_\_add\_\_(other)

if self.min\_current == None:

self.min\_current = other.min\_current

if self.max\_temperature == None:

self.max\_temperature = other.max\_temperature

self.condition = {"current": ">" + str(self.min\_current),

"temperature": "<" + str(self.max\_temperature)}

return self

def \_\_iter\_\_(self):

return zip(self.\_get(self.get, self.condition).keys(),

list(values[0][0]

for values in self.\_get(self.get, self.condition).values()),

list(values[0][1]

for values in self.\_get(self.get, self.condition).values()),

list(values[1]

for values in self.\_get(self.get, self.condition).values()),

list(values[2]

for values in self.\_get(self.get, self.condition).values()))

if \_\_name\_\_ == '\_\_main\_\_':

import os

import sys

from task6\_module import run

data = Datasource()

dirname = sys.argv[1] if len(sys.argv) > 1 else None

for filename in sorted(os.listdir(dirname or '.')):

if filename.startswith('data') and filename.endswith('.txt'):

data += Datasource(filename, min\_current=20, max\_temperature=40)

for timemark, temperature, current, probe\_1, probe\_2 in data:

print('Time = {:3d}, Temperature = {:2d}, Current = {:2d}, Probe\_1 = {:2d}, Probe\_2 = {:2d}'.format(

timemark, temperature, current, probe\_1, probe\_2))

pass