

# CI4 ソフトウェア実験

CI4 21 番 下石 龍生

2021 年 5 月 7 日

## 1 目的

本実験は、Script 言語である Python の基礎を身につけることを目的とする。まず、ファイルの入出力を習得し、今後の研究、実験で活用できるようにする。

## 2 Python とは

Python<sup>[2]</sup>とは分散オペレーティングシステム「Amoeba」のシステム管理を行うために Guido van Rosseum 氏が開発したプログラミング言語である。インタプリタ型言語であり、読みやすく、エラーの修正がしやすいため、初心者でも扱いやすい言語である。Python が活用されている例を以下に示す。

- データの収集、処理、分析
- クローリング、スクレイピング
- 機械学習、ディープラーニング
- ゲーム、スマホアプリの開発
- Web アプリケーション開発

## 3 実験環境

実験環境を以下の表 1 に示す。

表 1 実験環境

デバイス	OS	ソフト
MacBookAir2019 13inch	MacOS BigSur 11.2.3	Python3.9.4

## 4 課題

### 4.1 データセット

本実験で使用したデータを以下の Code1, 2, 3, 4, 5 に示す。

Code 1 input.csv

```
1 1, 0.1, 5, 1st
2 2, 0.8, 8, 2nd
3 3, 0.8, 7, 3rd
4 4, 0.2, 2, 4th
```

Code 2 input2.csv

```
1 ## Comment for date
2 1, 0.1, 5, 1st
3 2, 0.8, 8, 2nd
4 3, 0.8, 7, 3rd
5 4, 0.2, 2, 4th
```

## Code 3 data1.csv

---

```

1 78.27, 2.2433E-7, 9.9996E+0, -9.9992E+0, 9.90672E-3, -1.63474E-2, 1.3128E-3, 4.6242E-7, -1.8707E+35
2 78.44, 2.2475E-7, 9.9996E+0, -9.9992E+0, 9.98927E-3, -1.63138E-2, 1.3152E-3, 5.2116E-7, -1.9507E+35
3 86.45, 2.8084E-7, 9.9996E+0, -9.9992E+0, 1.40467E-2, -1.88213E-2, 1.6435E-3, 4.2089E-7, -1.9292E+35
4 87.55, 2.9212E-7, 9.9996E+0, -9.9992E+0, 1.43470E-2, -1.98409E-2, 1.7095E-3, 4.4284E-7, -1.8769E+35
5 90.82, 3.1456E-7, 9.9996E+0, -9.9992E+0, 1.57977E-2, -2.10160E-2, 1.8408E-3, 2.2147E-7, -1.8569E+35
6 89.14, 3.0312E-7, 9.9996E+0, -9.9992E+0, 1.63984E-2, -1.90774E-2, 1.7739E-3, 1.2772E-6, -1.8153E+35
7 90.50, 3.1425E-7, 9.9996E+0, -9.9992E+0, 1.68736E-2, -1.99045E-2, 1.8390E-3, 2.6109E-6, -1.8646E+35
8 90.92, 3.1701E-7, 9.9996E+0, -9.9992E+0, 1.76287E-2, -1.94725E-2, 1.8552E-3, 4.3949E-7, -1.8569E+35
9 89.72, 3.1371E-7, 9.9996E+0, -9.9992E+0, 1.72048E-2, -1.95096E-2, 1.8358E-3, 1.5013E-6, -1.8799E+35
10 91.88, 3.1787E-7, 9.9996E+0, -9.9992E+0, 1.55739E-2, -2.16276E-2, 1.8602E-3, 1.2286E-5, -1.8646E+35
11 92.31, 3.2427E-7, 9.9996E+0, -9.9992E+0, 1.56855E-2, -2.22650E-2, 1.8976E-3, 1.7993E-7, -1.8092E+35
12 92.94, 3.2919E-7, 9.9996E+0, -9.9992E+0, 1.60052E-2, -2.25214E-2, 1.9264E-3, 2.8432E-6, -1.8323E+35
13 93.46, 3.3335E-7, 9.9996E+0, -9.9992E+0, 1.62791E-2, -2.27337E-2, 1.9508E-3, 4.0930E-7, -1.8184E+35
14 94.26, 3.3884E-7, 9.9996E+0, -9.9992E+0, 1.71364E-2, -2.25196E-2, 1.9829E-3, 3.8164E-7, -1.8199E+35
15 94.50, 3.3998E-7, 9.9996E+0, -9.9992E+0, 1.74291E-2, -2.23597E-2, 1.9896E-3, 2.9308E-7, -1.8676E+35
16 94.68, 3.4130E-7, 9.9996E+0, -9.9992E+0, 1.92182E-2, -2.07249E-2, 1.9973E-3, 1.8772E-6, -1.7953E+35
17 95.85, 3.5021E-7, 9.9996E+0, -9.9992E+0, 2.02971E-2, -2.06897E-2, 2.0495E-3, 4.1406E-6, -1.7599E+35
18 97.43, 3.6255E-7, 9.9996E+0, -9.9992E+0, 1.88141E-2, -2.36159E-2, 2.1216E-3, 2.1107E-7, -1.7861E+35
19 97.73, 3.6476E-7, 9.9996E+0, -9.9992E+0, 1.84883E-2, -2.42011E-2, 2.1346E-3, 5.0817E-7, -1.7676E+35
20 98.56, 3.6910E-7, 9.9996E+0, -9.9992E+0, 1.85176E-2, -2.46798E-2, 2.1600E-3, 4.3085E-7, -1.8215E+35
21 102.25, 3.9683E-7, 9.9996E+0, -9.9992E+0, 2.02126E-2, -2.62293E-2, 2.3222E-3, 1.7037E-7, -1.7430E+35
22 105.50, 4.2004E-7, 9.9996E+0, -9.9993E+0, 2.13266E-2, -2.78318E-2, 2.4581E-3, 2.6166E-7, -1.7384E+35
23 110.53, 4.5810E-7, 9.9996E+0, -9.9992E+0, 2.38603E-2, -2.97530E-2, 2.6808E-3, 2.3523E-7, -1.7876E+35
24 115.53, 4.9537E-7, 9.9996E+0, -9.9992E+0, 2.62402E-2, -3.17345E-2, 2.8989E-3, 5.5369E-7, -1.7507E+35
25 120.38, 5.1760E-7, 9.9996E+0, -9.9992E+0, 3.06920E-2, -2.98848E-2, 3.0290E-3, 5.5000E-6, -1.7338E+35
26 125.50, 5.7097E-7, 9.9996E+0, -9.9992E+0, 3.10459E-2, -3.57768E-2, 3.3413E-3, 3.1731E-7, -1.7492E+35
27 129.53, 6.0008E-7, 9.9996E+0, -9.9992E+0, 3.25195E-2, -3.77102E-2, 3.5117E-3, 1.7189E-6, -1.7584E+35
28 135.40, 6.4383E-7, 9.9996E+0, -9.9993E+0, 3.52213E-2, -4.01288E-2, 3.7677E-3, 4.4367E-7, -1.7015E+35
29 140.26, 6.7973E-7, 9.9996E+0, -9.9993E+0, 3.69739E-2, -4.25774E-2, 3.9778E-3, 7.9899E-7, -1.7015E+35
30 148.37, 7.3960E-7, 9.9996E+0, -9.9993E+0, 4.03453E-2, -4.62123E-2, 4.3281E-3, 6.5347E-7, -1.6615E+35
31 156.15, 7.9678E-7, 9.9996E+0, -9.9993E+0, 4.34485E-2, -4.98013E-2, 4.6628E-3, 2.2521E-6, -1.6584E+35
32 164.15, 8.5599E-7, 9.9996E+0, -9.9993E+0, 4.69368E-2, -5.32432E-2, 5.0093E-3, 5.3531E-7, -1.6538E+35
33 172.26, 9.1971E-7, 9.9996E+0, -9.9993E+0, 5.15151E-2, -5.61215E-2, 5.3821E-3, 4.3543E-6, -1.6123E+35
34 180.39, 9.7510E-7, 9.9996E+0, -9.9993E+0, 5.41348E-2, -5.99845E-2, 5.7063E-3, 6.9338E-7, -1.6492E+35
35 186.24, 1.0210E-6, 9.9996E+0, -9.9993E+0, 5.89843E-2, -6.05116E-2, 5.9751E-3, 2.3953E-7, -1.5230E+35
36 196.17, 1.0890E-6, 9.9996E+0, -9.9993E+0, 6.06092E-2, -6.68363E-2, 6.3726E-3, 5.5510E-7, -1.3276E+35
37 204.14, 1.1467E-6, 9.9996E+0, -9.9993E+0, 6.36906E-2, -7.05126E-2, 6.7105E-3, 5.2527E-7, -1.3353E+35
38 211.83, 1.2025E-6, 9.9996E+0, -9.9993E+0, 6.64436E-2, -7.42855E-2, 7.0369E-3, 7.4563E-7, -1.2692E+35
39 220.06, 1.2620E-6, 9.9996E+0, -9.9993E+0, 7.00927E-2, -7.76014E-2, 7.3851E-3, 1.2198E-6, -1.2476E+35
40 228.02, 1.3188E-6, 9.9996E+0, -9.9993E+0, 7.34569E-2, -8.08883E-2, 7.7177E-3, 8.5385E-7, -1.1984E+35
41 236.13, 1.3769E-6, 9.9996E+0, -9.9993E+0, 7.67535E-2, -8.43907E-2, 8.0577E-3, 6.1389E-7, -1.2369E+35
42 243.94, 1.4329E-6, 9.9996E+0, -9.9993E+0, 7.98894E-2, -8.78139E-2, 8.3856E-3, 8.7021E-7, -1.1369E+35
43 251.97, 1.4900E-6, 9.9996E+0, -9.9993E+0, 8.32121E-2, -9.11665E-2, 8.7194E-3, 9.4080E-7, -1.1492E+35
44 259.53, 1.5437E-6, 9.9996E+0, -9.9993E+0, 8.63381E-2, -9.43322E-2, 9.0340E-3, 8.0184E-7, -1.1015E+35
45 267.57, 1.6022E-6, 9.9995E+0, -9.9993E+0, 9.04222E-2, -9.70898E-2, 9.3761E-3, 6.1068E-7, -1.0353E+35
46 276.04, 1.6609E-6, 9.9995E+0, -9.9993E+0, 9.38270E-2, -1.00550E-1, 9.7194E-3, 1.6502E-6, -1.0230E+35
47 283.99, 1.7105E-6, 9.9996E+0, -9.9993E+0, 9.95446E-2, -1.00647E-1, 1.0010E-2, 2.9944E-6, -1.1630E+35

```

---

## Code 4 data2.csv

---

```

1 ## http://sheat.phys.tohoku.ac.jp/~tmatsu/demae/demae.shtml
2 ## Temp(K), rho(Ohm*cm), I+(mA), I-(mA), V+(mV), V-(mV), R(Ohm), errR(Ohm), Rprobe(Ohm)
3 78.40, 8.0298E+2, 9.9997E-4, -9.9992E-4, 1.11027E+1, -1.10989E+1, 1.1101E+4, 5.2611E+3, -2.4864E+39
4 78.32, 8.0423E+2, 9.9997E-4, -9.9992E-4, 1.11207E+1, -1.11156E+1, 1.1119E+4, 3.3048E+3, -2.4807E+39
5 86.17, 4.0653E+2, 9.9997E-4, -9.9992E-4, 5.48185E+0, -5.75847E+0, 5.6205E+3, 6.9381E+4, -2.4713E+39
6 87.07, 3.3478E+2, 9.9997E-4, -9.9992E-4, 4.64757E+0, -4.60877E+0, 4.6284E+3, 8.4885E+3, -2.4732E+39
7 91.03, 2.2032E+2, 9.9997E-4, -9.9992E-4, 3.04943E+0, -3.04217E+0, 3.0460E+3, 5.2593E+2, -2.5145E+39
8 89.59, 2.2537E+2, 9.9997E-4, -9.9992E-4, 3.15517E+0, -3.07600E+0, 3.1157E+3, 7.0777E+3, -2.5446E+39
9 89.69, 2.7721E+2, 9.9997E-4, -9.9992E-4, 3.96323E+0, -3.70135E+0, 3.8325E+3, 3.4595E+4, -2.4789E+39
10 91.08, 2.1688E+2, 9.9997E-4, -9.9992E-4, 2.99044E+0, -3.00609E+0, 2.9984E+3, 2.7848E+3, -2.502E+39
11 90.98, 2.1850E+2, 9.9997E-4, -9.9992E-4, 3.05741E+0, -2.98386E+0, 3.0208E+3, 6.7896E+3, -2.5089E+39
12 91.40, 2.2405E+2, 9.9997E-4, -9.9992E-4, 3.08848E+0, -3.10641E+0, 3.0976E+3, 1.2353E+4, -2.4789E+39
13 92.22, 1.9501E+2, 9.9997E-4, -9.9992E-4, 2.69954E+0, -2.69220E+0, 2.6960E+3, 8.1609E+2, -2.4451E+39
14 92.60, 1.8899E+2, 9.9997E-4, -9.9992E-4, 2.61212E+0, -2.61334E+0, 2.6129E+3, 6.1681E+2, -2.4488E+39
15 93.28, 1.7818E+2, 9.9997E-4, -9.9992E-4, 2.46033E+0, -2.46619E+0, 2.4634E+3, 9.1906E+2, -2.4451E+39
16 94.12, 1.6423E+2, 9.9997E-4, -9.9992E-4, 2.26580E+0, -2.27496E+0, 2.2705E+3, 1.4600E+3, -2.4244E+39
17 94.51, 1.5786E+2, 9.9996E-2, -9.9992E-2, 2.18155E+2, -2.18299E+2, 2.1824E+3, 1.9232E+2, -1.5204E+37
18 94.62, 1.5827E+2, 9.9996E-2, -9.9992E-2, 2.19717E+2, -2.17881E+2, 2.1881E+3, 1.3885E+3, -1.5528E+37
19 95.15, 1.6288E+2, 9.9995E-2, -9.9992E-2, 2.26925E+2, -2.23409E+2, 2.2518E+3, 4.4722E+3, -1.5012E+37
20 97.38, 1.2021E+2, 9.9996E-2, -9.9992E-2, 1.65938E+2, -1.66429E+2, 1.6619E+3, 2.9885E+2, -1.5348E+37
21 97.56, 1.1735E+2, 9.9996E-2, -9.9992E-2, 1.62509E+2, -1.61954E+2, 1.6224E+3, 3.4813E+2, -1.5528E+37
22 98.19, 1.1290E+2, 9.9995E-2, -9.9992E-2, 1.56012E+2, -1.56146E+2, 1.5609E+3, 3.4693E+2, -1.5156E+37
23 101.69, 8.4240E+1, 9.9995E-2, -9.9992E-2, 1.15619E+2, -1.17297E+2, 1.1646E+3, 7.0860E+2, -1.4988E+37
24 104.74, 6.5790E+1, 9.9995E-2, -9.9992E-2, 9.04867E+1, -9.14156E+1, 9.0957E+2, 3.4159E+2, -1.4903E+37
25 109.73, 4.4407E+1, 9.9995E-2, -9.9992E-2, 6.10731E+1, -6.17064E+1, 6.1393E+2, 1.3804E+2, -1.4819E+37
26 114.72, 3.0567E+1, 9.9995E-2, -9.9992E-2, 4.20374E+1, -4.24777E+1, 4.2260E+2, 7.1866E+1, -1.4879E+37
27 119.36, 2.2609E+1, 9.9995E-2, -9.9992E-2, 3.11887E+1, -3.13238E+1, 3.1258E+2, 1.6750E+1, -1.4795E+37
28 124.82, 1.5915E+1, 9.9995E-2, -9.9993E-2, 2.18804E+1, -2.21237E+1, 2.2003E+2, 1.8502E+1, -1.4315E+37
29 128.85, 1.2502E+1, 9.9995E-1, -9.9992E-1, 1.73125E+2, -1.72537E+2, 1.7284E+2, 1.9601E+0, -1.5057E+36
30 134.51, 9.2892E+0, 9.9995E-1, -9.9993E-1, 1.27961E+2, -1.28876E+2, 1.2843E+2, 4.0851E+0, -1.4519E+36
31 139.25, 7.3474E+0, 9.9995E-1, -9.9992E-1, 1.01271E+2, -1.01878E+2, 1.0158E+2, 2.3224E+0, -1.4596E+36
32 147.15, 5.1473E+0, 9.9995E-1, -9.9992E-1, 7.08655E+1, -7.14519E+1, 7.1163E+1, 1.5967E+0, -1.4634E+36
33 154.81, 3.7623E+0, 9.9995E-1, -9.9993E-1, 5.18527E+1, -5.21700E+1, 5.2014E+1, 6.1972E-1, -1.4288E+36

```

---

```

34 162.97, 2.8227E+0, 9.9995E-1, -9.9993E-1, 3.88986E+1, -3.91471E+1, 3.9025E+1, 3.4953E-1, -1.3903E+36
35 170.84, 2.2345E+0, 9.9995E-1, -9.9993E-1, 3.07780E+1, -3.10036E+1, 3.0893E+1, 2.7743E-1, -1.4038E+36
36 178.99, 1.7775E+0, 9.9995E-1, -9.9993E-1, 2.44901E+1, -2.46556E+1, 2.4574E+1, 1.4374E-1, -1.3577E+36
37 186.39, 1.4553E+0, 9.9995E-1, -9.9993E-1, 2.01453E+1, -2.00917E+1, 2.0120E+1, 2.8502E-2, -1.3403E+36
38 194.50, 1.2288E+0, 9.9995E-1, -9.9993E-1, 1.69324E+1, -1.70436E+1, 1.6989E+1, 6.6270E-2, -1.0730E+36
39 202.49, 1.0554E+0, 9.9995E-1, -9.9993E-1, 1.45430E+1, -1.46374E+1, 1.4591E+1, 4.9629E-2, -1.0211E+36
40 209.95, 9.3090E-1, 9.9995E-1, -9.9993E-1, 1.28323E+1, -1.29060E+1, 1.2870E+1, 3.2439E-2, -1.0327E+36
41 218.07, 8.2413E-1, 9.9995E-1, -9.9993E-1, 1.13565E+1, -1.14300E+1, 1.1394E+1, 3.0753E-2, -1.0057E+36
42 226.23, 7.3532E-1, 9.9995E-1, -9.9993E-1, 1.01415E+1, -1.01894E+1, 1.0166E+1, 1.9626E-2, -1.0134E+36
43 234.30, 6.6771E-1, 9.9995E-1, -9.9993E-1, 9.20993E+0, -9.25153E+0, 9.2313E+0, 1.4565E-2, -9.8459E+35
44 241.97, 6.1463E-1, 9.9995E-1, -9.9993E-1, 8.48039E+0, -8.51353E+0, 8.4975E+0, 1.1253E-2, -9.0959E+35
45 250.09, 5.7068E-1, 9.9995E-1, -9.9993E-1, 7.87407E+0, -7.90463E+0, 7.8898E+0, 8.6991E-3, -9.3651E+35
46 257.84, 5.3634E-1, 9.9995E-1, -9.9993E-1, 7.40344E+0, -7.42597E+0, 7.4151E+0, 6.5465E-3, -8.5959E+35
47 266.00, 5.0495E-1, 9.9995E-1, -9.9993E-1, 6.97268E+0, -6.98881E+0, 6.9812E+0, 4.6907E-3, -8.4228E+35
48 274.34, 4.7864E-1, 9.9995E-1, -9.9993E-1, 6.61300E+0, -6.62089E+0, 6.6173E+0, 2.8061E-3, -7.9613E+35
49 282.51, 4.5803E-1, 9.9995E-1, -9.9993E-1, 6.33378E+0, -6.33025E+0, 6.3324E+0, 3.7458E-3, -8.4613E+35

```

#### Code 5 data3.csv

```

1 ## http://sheat.phys.tohoku.ac.jp/~tmatsu/demae/demae.shtml
2 ## Temp(K), rho(Ohm*cm), I+(mA), I-(mA), V+(mV), V-(mV), R(Ohm), errR(Ohm), Rprobe(Ohm)
3 78.35, 8.4988E-7, 1.9989E+0, -1.9987E+0, -3.00004E-3, -3.02083E-3, 5.2003E-6, 1.4030E-6, -2.0839E+36
4 78.31, -4.7321E-6, 1.9989E+0, -1.9987E+0, -2.68194E-3, -2.56619E-3, -2.8955E-5, 4.0052E-6, -2.0665E+36
5 86.32, -2.4543E-6, 1.9989E+0, -1.9987E+0, -3.65151E-3, -3.59147E-3, -1.5018E-5, 8.5440E-6, -2.1166E+36
6 87.32, 8.7469E-6, 1.9989E+0, -1.9987E+0, -3.32761E-3, -3.54157E-3, 5.3521E-5, 1.8728E-6, -2.0531E+36
7 90.91, 1.1971E-5, 1.9989E+0, -1.9987E+0, -3.73067E-3, -4.02349E-3, 7.3249E-5, 9.3120E-7, -1.9934E+36
8 89.32, 1.2290E-5, 1.9989E+0, -1.9987E+0, -4.19682E-3, -4.49744E-3, 7.5200E-5, 3.6934E-6, -2.0367E+36
9 90.03, 1.5006E-5, 1.9989E+0, -1.9987E+0, -3.65441E-3, -4.02146E-3, 9.1817E-5, 1.2646E-6, -1.9251E+36
10 91.04, -9.4812E-6, 1.9989E+0, -1.9987E+0, -3.61564E-3, -3.38372E-3, -5.8014E-5, 2.1003E-5, -2.0742E+36
11 89.76, 1.2979E-5, 1.9989E+0, -1.9987E+0, -3.26653E-3, -3.58400E-3, 7.9414E-5, 4.4758E-6, -1.9895E+36
12 91.65, 2.0989E-5, 1.9989E+0, -1.9987E+0, -3.15513E-3, -3.66854E-3, 1.2843E-4, 1.3700E-5, -1.9048E+36
13 92.33, 4.9946E-5, 1.9989E+0, -1.9987E+0, -3.10190E-3, -4.32364E-3, 3.0561E-4, 8.5386E-7, -2.0001E+36
14 92.75, 6.8950E-5, 1.9989E+0, -1.9987E+0, -2.97513E-3, -4.66172E-3, 4.2190E-4, 3.8610E-6, -1.9636E+36
15 93.37, 2.2276E-4, 1.9989E+0, -1.9988E+0, -1.13484E-3, -6.58373E-3, 1.3630E-3, 5.0790E-7, -1.8846E+36
16 94.18, 2.3378E-4, 1.9989E+0, -1.9988E+0, -5.90276E-4, -6.30890E-3, 1.4305E-3, 1.5380E-6, -1.8971E+36
17 94.48, 2.3744E-4, 1.9989E+0, -1.9988E+0, 5.65277E-4, -5.24287E-3, 1.4529E-3, 1.9513E-6, -1.9087E+36
18 94.62, 2.0868E-4, 1.9989E+0, -1.9988E+0, 3.29340E-4, -4.77534E-3, 1.2769E-3, 9.9065E-6, -1.9270E+36
19 95.29, 2.4723E-4, 1.9989E+0, -1.9988E+0, 8.22104E-4, -5.22536E-3, 1.5128E-3, 1.8584E-5, -1.8298E+36
20 97.45, 2.5083E-4, 1.9989E+0, -1.9988E+0, 3.15718E-4, -5.82000E-3, 1.5348E-3, 1.6598E-6, -1.8856E+36
21 97.68, 2.4950E-4, 1.9989E+0, -1.9988E+0, 4.38839E-4, -5.66420E-3, 1.5267E-3, 6.2054E-7, -1.9385E+36
22 98.39, 2.4606E-4, 1.9989E+0, -1.9988E+0, 1.02077E-4, -5.91684E-3, 1.5056E-3, 3.2587E-6, -1.8192E+36
23 102.00, 2.6751E-4, 1.9989E+0, -1.9988E+0, 8.74538E-5, -6.45625E-3, 1.6369E-3, 1.4422E-6, -1.7701E+36
24 105.14, 2.8102E-4, 1.9989E+0, -1.9988E+0, 9.68370E-5, -6.77731E-3, 1.7195E-3, 7.7957E-6, -1.7239E+36
25 110.10, 3.0117E-4, 1.9989E+0, -1.9988E+0, 5.65670E-4, -6.80141E-3, 1.8428E-3, 1.1915E-5, -1.7499E+36
26 115.11, 3.2225E-4, 1.9989E+0, -1.9988E+0, 9.12772E-4, -6.96986E-3, 1.9718E-3, 3.2484E-6, -1.7470E+36
27 119.75, 3.4881E-4, 1.9989E+0, -1.9988E+0, 2.61654E-3, -5.91582E-3, 2.1343E-3, 2.0617E-5, -1.6584E+36
28 125.15, 3.6479E-4, 1.9989E+0, -1.9988E+0, 1.76919E-3, -7.15398E-3, 2.2321E-3, 9.3778E-7, -1.6690E+36
29 129.15, 3.7386E-4, 1.9989E+0, -1.9988E+0, 1.80181E-3, -7.34324E-3, 2.2876E-3, 1.8753E-6, -1.6565E+36
30 134.92, 4.0593E-4, 1.9989E+0, -1.9988E+0, 2.33490E-3, -7.59457E-3, 2.4838E-3, 1.2696E-6, -1.5217E+36
31 139.75, 4.1951E-4, 1.9989E+0, -1.9988E+0, 2.23826E-3, -8.02352E-3, 2.5669E-3, 6.2759E-7, -1.6122E+36
32 147.75, 4.5178E-4, 1.9989E+0, -1.9988E+0, 2.59347E-3, -8.45766E-3, 2.7644E-3, 4.1930E-6, -1.5323E+36
33 155.42, 4.7606E-4, 1.9989E+0, -1.9988E+0, 3.01220E-3, -8.63273E-3, 2.9129E-3, 2.7603E-6, -1.4881E+36
34 163.50, 5.0517E-4, 1.9989E+0, -1.9988E+0, 3.53376E-3, -8.82327E-3, 3.0911E-3, 7.3491E-7, -1.3889E+36
35 171.65, 5.3828E-4, 1.9989E+0, -1.9988E+0, 4.78093E-3, -8.38609E-3, 3.2937E-3, 3.7993E-6, -1.4659E+36
36 179.76, 5.6113E-4, 1.9989E+0, -1.9988E+0, 4.41246E-3, -9.31358E-3, 3.4335E-3, 1.1255E-6, -1.2821E+36
37 186.30, 5.8351E-4, 1.9989E+0, -1.9988E+0, 6.42218E-3, -7.85123E-3, 3.5704E-3, 3.9036E-7, -1.2012E+36
38 195.32, 6.0834E-4, 1.9989E+0, -1.9988E+0, 5.31360E-3, -9.56733E-3, 3.7224E-3, 1.0203E-6, -6.1504E+35
39 203.31, 6.2953E-4, 1.9989E+0, -1.9988E+0, 5.65524E-3, -9.74403E-3, 3.8520E-3, 1.7085E-6, -4.9666E+35
40 210.85, 6.4588E-4, 1.9989E+0, -1.9988E+0, 5.80046E-3, -9.99862E-3, 3.9521E-3, 2.1140E-6, -3.8404E+35
41 219.08, 6.7698E-4, 1.9989E+0, -1.9988E+0, 6.59176E-3, -9.96802E-3, 4.1423E-3, 2.8030E-6, -3.7634E+35
42 227.12, 6.9838E-4, 1.9989E+0, -1.9988E+0, 6.66789E-3, -1.04154E-2, 4.2733E-3, 8.8692E-6, -3.3399E+35
43 235.21, 7.1719E-4, 1.9989E+0, -1.9988E+0, 6.93940E-3, -1.06040E-2, 4.3884E-3, 7.9470E-7, -3.3495E+35
44 242.90, 7.3836E-4, 1.9989E+0, -1.9988E+0, 7.27420E-3, -1.07872E-2, 4.5180E-3, 1.2676E-6, -1.8384E+35
45 251.10, 7.5852E-4, 1.9989E+0, -1.9988E+0, 7.56648E-3, -1.09881E-2, 4.6413E-3, 1.1781E-6, -1.6459E+35
46 258.69, 7.7807E-4, 1.9988E+0, -1.9988E+0, 7.87015E-3, -1.11625E-2, 4.7609E-3, 7.3464E-7, -1.7325E+34
47 266.71, 7.9811E-4, 1.9988E+0, -1.9988E+0, 8.29933E-3, -1.12235E-2, 4.8835E-3, 1.3722E-6, 1.0203E+35
48 275.15, 8.3132E-4, 1.9988E+0, -1.9989E+0, 1.03827E-2, -9.95254E-3, 5.0867E-3, 1.3586E-5, 2.9741E+35
49 283.23, 8.7133E-4, 1.9988E+0, -1.9988E+0, 1.19352E-2, -9.37879E-3, 5.3316E-3, 1.2972E-5, 0.0000E+0

```

## 4.2 演習課題 1

30 以下の偶数の 2 乗の和を計算・表示する script を作成せよ。

作成したコードを以下の Code6 に、その実行結果を図 1 に示す。

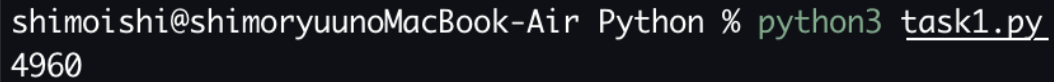
#### Code 6 演習課題 1

```

1 ans = 0
2
3 for i in range(0, 31, 2):

```

```
4     ans += i ** 2
5
6 print(ans)
```



```
shimoishi@shimoryuunoMacBook-Air Python % python3 task1.py
4960
```

図1 演習課題1の実行結果

演習課題1では、for文で0 31まで、2つ間隔で値をiに代入しているため偶数だけを取り除くことができるようにしている。取り出した偶数iは、二乗して変数ansに足して更新するため、結果的に30以下の偶数の二乗和が計算されている。

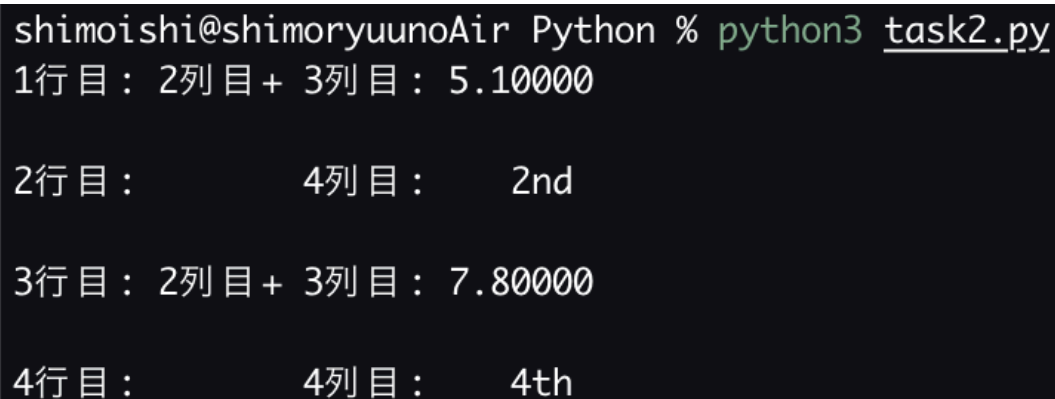
### 4.3 演習課題2

input.csvを読み込んで、図12のように、奇数行の場合は行数および2列目+3列目を浮動小数型で、偶数行の場合は行数および4列目を文字型で、画面に表示するscriptを作成せよ。

作成したコードをCode7に、その実行結果を図2に示す。なお、Code1を使用した。

Code 7 演習課題2

```
1 input = "input.csv"
2
3 FD = open(input, "r")
4
5 for line in FD:
6     line = line.rstrip("\n")
7     line = line.rstrip("\r")
8
9     tmp = line.split(",")
10    i = int(tmp[0])
11    j = tmp[3]
12    if i % 2 == 0:
13        print("%d行目:      4列目: %6s\n" % (i, j))
14    else:
15        print("%d行目: 2列目+ 3列目: %.5f\n" % (i, float(tmp[1])+float(tmp[2])))
16 FD.close()
```



```
shimoishi@shimoryuunoAir Python % python3 task2.py
1行目: 2列目+ 3列目: 5.10000

2行目:      4列目:      2nd

3行目: 2列目+ 3列目: 7.80000

4行目:      4列目:      4th
```

図2 演習課題2の実行結果

演習課題2では、input.csvを1行ごとに取り出してそれをさらに列ごとに配列の要素化している。配列にするときは","区切りで分けた。行番号を2で割ったときに割り切れたときは偶数行、割り切れなかったときは奇数行として表示する形式を変更できるようにしている。

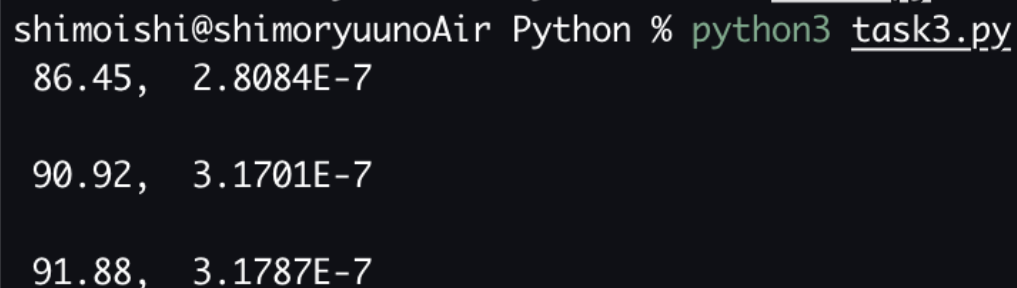
## 4.4 演習課題 3

まず data1.csv を保存し、data1.csv の 1, 2 行目のコメント行を Editor などで削除しなさい。このコメント行を削除したデータを読み込み、その 3, 8, 10 行目のみ、各行の 1, 2 列目を表示する script を作成せよ。(Script 例リスト 7 を参考にして、コメント行を削除せずに、その行を読み飛ばす処理を行っても良い。) さらに出力行を図 13 のようなリストで指定し、指定された行の 1, 2 列目を表示する script を作成せよ。ただしリスト loutput の要素数は 3 とは限らない。また len を用いるとリストの要素数を取得することができる。

この課題で作成したコードとその実行結果をそれぞれ Code8 と図 3 に示す。なお、Code3 を使用した。

Code 8 演習課題 3

```
1 input = "data1.csv"
2 line_count = 0
3 FD = open(input, "r")
4 loutput = [3, 8, 10]
5
6 for line in FD:
7     line_count += 1
8     line = line.rstrip("\n")
9     line = line.rstrip("\r")
10
11     tmp = line.split(",")
12     if line_count in loutput:
13         print("%6s, %6s\n" %(tmp[0], tmp[1]))
14
15 FD.close()
```



```
shimoishi@shimoryuunoAir Python % python3 task3.py
86.45, 2.8084E-7

90.92, 3.1701E-7

91.88, 3.1787E-7
```

図 3 演習課題 3 の実行結果

## 4.5 演習課題 4

前の例題で用いた csv ファイル data1.csv およびそれと同じフォーマットの csv ファイル data2.csv, csv ファイル data3.csv のそれぞれに対して、温度 (1 列目), 抵抗率 (2 列目), 電気抵抗 (7 列目), 電気抵抗の測定誤差 (8 列目) を抜き出し、ファイルに出力する script を作成せよ。ただし、抵抗値が負のデータに関しては、先頭に "##" を付けて出力すること。また出力ファイル名は適切に設定すること。

この課題で作成したコードを Code9 に示す。また、実行結果を Code10, 11, 11 に示す。なお、Code3, 4, 5 を使用した。

Code 9 演習課題 4

```
1 import re
2
3 input = ["data1.csv", "data2.csv", "data3.csv"]
4 output = ["data1.txt", "data2.txt", "data3.txt"]
5 err_max = 2e-6
6
7 ni = 100
8 nj = 9
9
10 for in_file, out_file in zip(input, output):
11     array = [[0 for i in range(ni)] for j in range(nj)]
```

```

12
13 FDI = open(in_file, "r")
14 FDO = open(out_file, "w")
15 nd = 0;
16 for line in FDI:
17     if (re.search(r"^\#", line)):
18         continue
19     line = line.rstrip("\n")
20     line = line.rstrip("\r")
21
22     tmp = line.split(",")
23     for j in range(nj):
24         array[j][nd] = float(tmp[j])
25     nd += 1
26
27 for i in range(nd):
28     T = array[0][i]
29     rho = array[1][i]
30     R = array[6][i]
31     err = array[7][i]
32
33     if (R < 0):
34         FDO.write("####")
35     FDO.write("%7.2f %11.4e %11.4e %11.4e\n" %(T, rho, R, err))
36 FDI.close()
37 FDO.close()

```

---

Code 10 出力 1

---

1	78.27	2.2433e-07	1.3128e-03	4.6242e-07
2	78.44	2.2475e-07	1.3152e-03	5.2116e-07
3	86.45	2.8084e-07	1.6435e-03	4.2089e-07
4	87.55	2.9212e-07	1.7095e-03	4.4284e-07
5	90.82	3.1456e-07	1.8408e-03	2.2147e-07
6	89.14	3.0312e-07	1.7739e-03	1.2772e-06
7	90.50	3.1425e-07	1.8390e-03	2.6109e-06
8	90.92	3.1701e-07	1.8552e-03	4.3949e-07
9	89.72	3.1371e-07	1.8358e-03	1.5013e-06
10	91.88	3.1787e-07	1.8602e-03	1.2286e-05
11	92.31	3.2427e-07	1.8976e-03	1.7993e-07
12	92.94	3.2919e-07	1.9264e-03	2.8432e-06
13	93.46	3.3335e-07	1.9508e-03	4.0930e-07
14	94.26	3.3884e-07	1.9829e-03	3.8164e-07
15	94.50	3.3998e-07	1.9896e-03	2.9308e-07
16	94.68	3.4130e-07	1.9973e-03	1.8772e-06
17	95.85	3.5021e-07	2.0495e-03	4.1406e-06
18	97.43	3.6255e-07	2.1216e-03	2.1107e-07
19	97.73	3.6476e-07	2.1346e-03	5.0817e-07
20	98.56	3.6910e-07	2.1600e-03	4.3085e-07
21	102.25	3.9683e-07	2.3222e-03	1.7037e-07
22	105.50	4.2004e-07	2.4581e-03	2.6166e-07
23	110.53	4.5810e-07	2.6808e-03	2.3523e-07
24	115.53	4.9537e-07	2.8989e-03	5.5369e-07
25	120.38	5.1760e-07	3.0290e-03	5.5000e-06
26	125.50	5.7097e-07	3.3413e-03	3.1731e-07
27	129.53	6.0008e-07	3.5117e-03	1.7189e-06
28	135.40	6.4383e-07	3.7677e-03	4.4367e-07
29	140.26	6.7973e-07	3.9778e-03	7.9899e-07
30	148.37	7.3960e-07	4.3281e-03	6.5347e-07
31	156.15	7.9678e-07	4.6628e-03	2.2521e-06
32	164.15	8.5599e-07	5.0093e-03	5.3531e-07
33	172.26	9.1971e-07	5.3821e-03	4.3543e-06
34	180.39	9.7510e-07	5.7063e-03	6.9338e-07
35	186.24	1.0210e-06	5.9751e-03	2.3953e-07
36	196.17	1.0890e-06	6.3726e-03	5.5510e-07
37	204.14	1.1467e-06	6.7105e-03	5.2527e-07
38	211.83	1.2025e-06	7.0369e-03	7.4563e-07
39	220.06	1.2620e-06	7.3851e-03	1.2198e-06
40	228.02	1.3188e-06	7.7177e-03	8.5385e-07
41	236.13	1.3769e-06	8.0577e-03	6.1389e-07
42	243.94	1.4329e-06	8.3856e-03	8.7021e-07
43	251.97	1.4900e-06	8.7194e-03	9.4080e-07
44	259.53	1.5437e-06	9.0340e-03	8.0184e-07
45	267.57	1.6022e-06	9.3761e-03	6.1068e-07
46	276.04	1.6609e-06	9.7194e-03	1.6502e-06
47	283.99	1.7105e-06	1.0010e-02	2.9944e-06

---

Code 11 出力 2

---

1	78.40	8.0298e+02	1.1101e+04	5.2611e+03
2	78.32	8.0423e+02	1.1119e+04	3.3048e+03
3	86.17	4.0653e+02	5.6205e+03	6.9381e+04
4	87.07	3.3478e+02	4.6284e+03	8.4885e+03
5	91.03	2.2032e+02	3.0460e+03	5.2593e+02

6	89.59	2.2537e+02	3.1157e+03	7.0777e+03
7	89.69	2.7721e+02	3.8325e+03	3.4595e+04
8	91.08	2.1688e+02	2.9984e+03	2.7848e+03
9	90.98	2.1850e+02	3.0208e+03	6.7896e+03
10	91.40	2.2405e+02	3.0976e+03	1.2353e+04
11	92.22	1.9501e+02	2.6960e+03	8.1609e+02
12	92.60	1.8899e+02	2.6129e+03	6.1681e+02
13	93.28	1.7818e+02	2.4634e+03	9.1906e+02
14	94.12	1.6423e+02	2.2705e+03	1.4600e+03
15	94.51	1.5786e+02	2.1824e+03	1.9232e+02
16	94.62	1.5827e+02	2.1881e+03	1.3885e+03
17	95.15	1.6288e+02	2.2518e+03	4.4722e+03
18	97.38	1.2021e+02	1.6619e+03	2.9885e+02
19	97.56	1.1735e+02	1.6224e+03	3.4813e+02
20	98.19	1.1290e+02	1.5609e+03	3.4693e+02
21	101.69	8.4240e+01	1.1646e+03	7.0860e+02
22	104.74	6.5790e+01	9.0957e+02	3.4159e+02
23	109.73	4.4407e+01	6.1393e+02	1.3804e+02
24	114.72	3.0567e+01	4.2260e+02	7.1866e+01
25	119.36	2.2609e+01	3.1258e+02	1.6750e+01
26	124.82	1.5915e+01	2.2003e+02	1.8502e+01
27	128.85	1.2502e+01	1.7284e+02	1.9601e+00
28	134.51	9.2892e+00	1.2843e+02	4.0851e+00
29	139.25	7.3474e+00	1.0158e+02	2.3224e+00
30	147.15	5.1473e+00	7.1163e+01	1.5967e+00
31	154.81	3.7623e+00	5.2014e+01	6.1972e-01
32	162.97	2.8227e+00	3.9025e+01	3.4953e-01
33	170.84	2.2345e+00	3.0893e+01	2.7743e-01
34	178.99	1.7775e+00	2.4574e+01	1.4374e-01
35	186.39	1.4553e+00	2.0120e+01	2.8502e-02
36	194.50	1.2288e+00	1.6989e+01	6.6270e-02
37	202.49	1.0554e+00	1.4591e+01	4.9629e-02
38	209.95	9.3090e-01	1.2870e+01	3.2439e-02
39	218.07	8.2413e-01	1.1394e+01	3.0753e-02
40	226.23	7.3532e-01	1.0166e+01	1.9626e-02
41	234.30	6.6771e-01	9.2313e+00	1.4565e-02
42	241.97	6.1463e-01	8.4975e+00	1.1253e-02
43	250.09	5.7068e-01	7.8898e+00	8.6991e-03
44	257.84	5.3634e-01	7.4151e+00	6.5465e-03
45	266.00	5.0495e-01	6.9812e+00	4.6907e-03
46	274.34	4.7864e-01	6.6173e+00	2.8061e-03
47	282.51	4.5803e-01	6.3324e+00	3.7458e-03

---

Code 12 出力 3

---

1	78.35	8.4988e-07	5.2003e-06	1.4030e-06	
2	###	78.31	-4.7321e-06	-2.8955e-05	4.0052e-06
3	###	86.32	-2.4543e-06	-1.5018e-05	8.5440e-06
4	87.32	8.7469e-06	5.3521e-05	1.8728e-06	
5	90.91	1.1971e-05	7.3249e-05	9.3120e-07	
6	89.32	1.2290e-05	7.5200e-05	3.6934e-06	
7	90.03	1.5006e-05	9.1817e-05	1.2646e-06	
8	###	91.04	-9.4812e-06	-5.8014e-05	2.1003e-05
9	89.76	1.2979e-05	7.9414e-05	4.4758e-06	
10	91.65	2.0989e-05	1.2843e-04	1.3700e-05	
11	92.33	4.9946e-05	3.0561e-04	8.5386e-07	
12	92.75	6.8950e-05	4.2190e-04	3.8610e-06	
13	93.37	2.2276e-04	1.3630e-03	5.0790e-07	
14	94.18	2.3378e-04	1.4305e-03	1.5380e-06	
15	94.48	2.3744e-04	1.4529e-03	1.9513e-06	
16	94.62	2.0868e-04	1.2769e-03	9.9065e-06	
17	95.29	2.4723e-04	1.5128e-03	1.8584e-05	
18	97.45	2.5083e-04	1.5348e-03	1.6598e-06	
19	97.68	2.4950e-04	1.5267e-03	6.2054e-07	
20	98.39	2.4606e-04	1.5056e-03	3.2587e-06	
21	102.00	2.6751e-04	1.6369e-03	1.4422e-06	
22	105.14	2.8102e-04	1.7195e-03	7.7957e-06	
23	110.10	3.0117e-04	1.8428e-03	1.1915e-05	
24	115.11	3.2225e-04	1.9718e-03	3.2484e-06	
25	119.75	3.4881e-04	2.1343e-03	2.0617e-05	
26	125.15	3.6479e-04	2.2321e-03	9.3778e-07	
27	129.15	3.7386e-04	2.2876e-03	1.8753e-06	
28	134.92	4.0593e-04	2.4838e-03	1.2696e-06	
29	139.75	4.1951e-04	2.5669e-03	6.2759e-07	
30	147.75	4.5178e-04	2.7644e-03	4.1930e-06	
31	155.42	4.7606e-04	2.9129e-03	2.7603e-06	
32	163.50	5.0517e-04	3.0911e-03	7.3491e-07	
33	171.65	5.3828e-04	3.2937e-03	3.7993e-06	
34	179.76	5.6113e-04	3.4335e-03	1.1255e-06	
35	186.30	5.8351e-04	3.5704e-03	3.9036e-07	
36	195.32	6.0834e-04	3.7224e-03	1.0203e-06	
37	203.31	6.2953e-04	3.8520e-03	1.7085e-06	
38	210.85	6.4588e-04	3.9521e-03	2.1140e-06	
39	219.08	6.7698e-04	4.1423e-03	2.8030e-06	
40	227.12	6.9838e-04	4.2733e-03	8.8692e-06	

41	235.21	7.1719e-04	4.3884e-03	7.9470e-07
42	242.90	7.3836e-04	4.5180e-03	1.2676e-06
43	251.10	7.5852e-04	4.6413e-03	1.1781e-06
44	258.69	7.7807e-04	4.7609e-03	7.3464e-07
45	266.71	7.9811e-04	4.8835e-03	1.3722e-06
46	275.15	8.3132e-04	5.0867e-03	1.3586e-05
47	283.23	8.7133e-04	5.3316e-03	1.2972e-05

---

## 4.6 研究課題

data1.csv, data2.csv, data3.csv に対して、以下の条件を満たすような script を作成せよ。

- 課題 4 の script で、matplotlib を使ってグラフを描画する。
- 横軸: 温度、縦軸: 電気抵抗値。
- 抵抗値の最大値・最小値に応じて、自動的に縦軸の範囲を指定。
- 最大値と最小値が二桁以上異なる場合は縦軸を対数グラフ。
- 画像 (png フォーマット) に保存。

研究課題で作成したコードを Code13 に示す。また、実行結果を図 4, 5, 6 にそれぞれ示す。なお、Code3, 4, 5 を使用した。

Code 13 研究課題

---

```
1 import matplotlib.pyplot as plt
2 import re
3
4 input = ["data1.csv", "data2.csv", "data3.csv"]
5 err_max = 2e-6
6
7 ni = 100
8 nj = 9
9
10 for in_file in input:
11     array = [[0 for i in range(ni)] for j in range(nj)]
12
13     FDI = open(in_file, "r")
14     nd = 0;
15     x = []
16     y = []
17
18     for line in FDI:
19         if (re.search(r"^\#", line)):
20             continue
21         line = line.rstrip("\n")
22         line = line.rstrip("\r")
23
24         tmp = line.split(",")
25         for j in range(nj):
26             array[j][nd] = float(tmp[j])
27         nd += 1
28
29     for i in range(nd):
30         T = array[0][i]
31         rho = array[1][i]
32         R = array[6][i]
33         err = array[7][i]
34         if float(R) > 0:
35             x.append(T)
36             y.append(R)
37
38     plt.plot(x,y,marker=".")
39     #print(min(y))
40     if max(y)/min(y) >= 100:
41         plt.yscale("log")
42     plt.xlabel("T(K)")
43     plt.ylabel("R(Ohm)")
44     plt.savefig(in_file+".png")
45     plt.clf()
46
47 FDI.close()
```

---



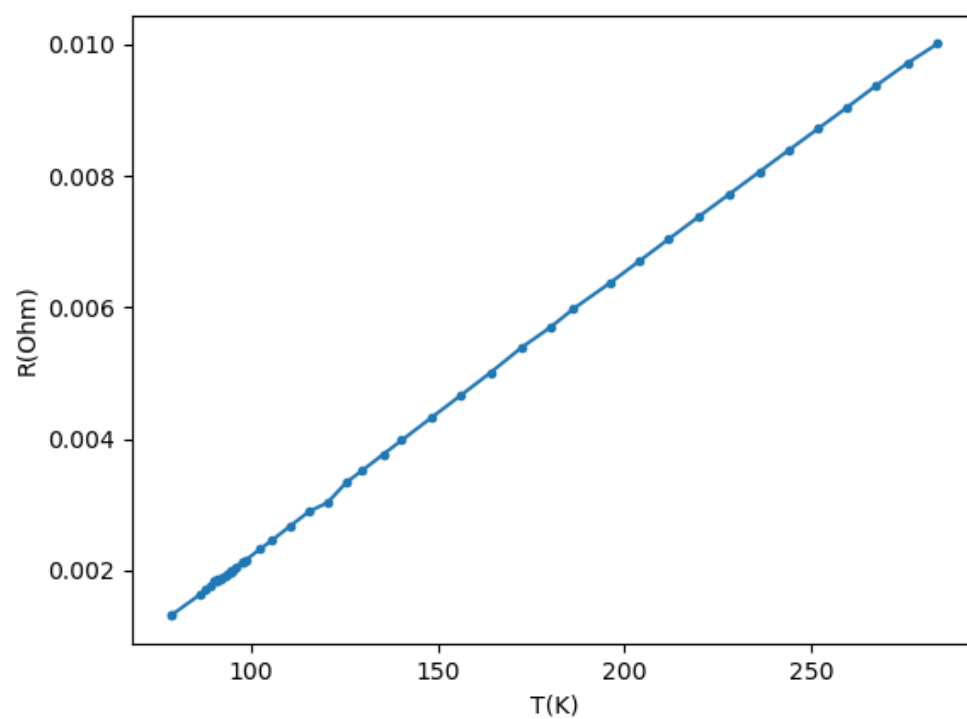


図4 data1.csv のグラフ

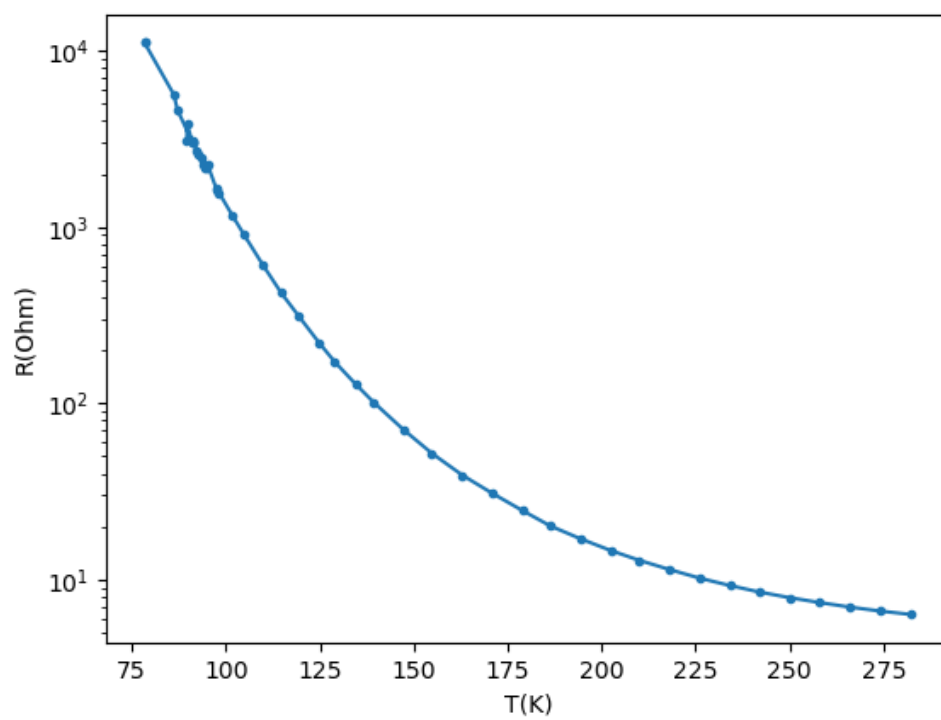


図5 data2.csv のグラフ

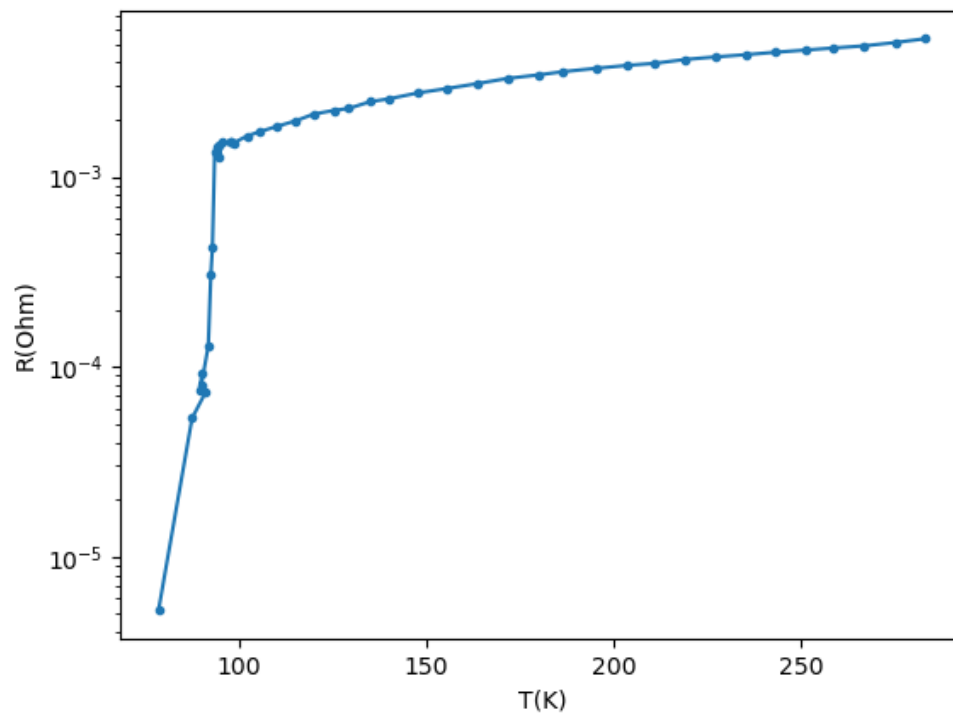


図6 data3.csv のグラフ

## 5 考察

今回の実験の課題 4 にて、読み込むファイルと書き出すファイルを対応付けて for 文で繰り返すプログラムを作成した。このとき使用した zip 関数について考察を行う。[1]

まず、2つの配列を対応付けて for 文を作る際に考えられるアルゴリズムを以下に示す。

---

```
1 a = ["n1", "n2", "n3"]
2 b = [1, 2, 3]
3 k = 0
4
5 ## パターン1
6 for i in a:
7     print(i, "=", str(b[k]))
8     k += 1
9
10 ## パターン2
11 for i in range(3):
12     print(a[i], "=", str(b[i]))
13
14 ## パターン3
15 for i,j in zip(a,b):
16     print(i,"=",str(j))
```

---

zip 関数を利用することにより、複数のリストを同時に所得し処理を行うことができるようになるため、他のアルゴリズムよりもシンプルに書くことができた。zip 関数で注意しなければ注意しなければならない点は、取得した 2 つ以上のリストで要素数が異なるときは、要素数の少ないほうに合わせて処理されることである。

## 6 感想

本実験では、プログラムをシンプルなアルゴリズムを使用して書くことができたため、スムーズに行うことができた。ファイルの入出力を行うことで今後の実験で効率よくデータ収集、集計を行うことができるようになる。これらを活用しながらさらに大きなデータ処理をできるようにさらに学習したい。

## 参考文献

- [1] Python "標準ライブラリ" (最終閲覧日 2021 年 5 月 7 日)  
<https://docs.python.org/ja/3/library/functions.html>

[2] @IT ”Python ってどんな言語なの？” (最終閲覧日 2021 年 5 月 7 日)

<https://www.atmarkit.co.jp/ait/articles/1904/02/news024.html>