

```

1 import os, sys
2 sys.path.append(os.path.join(os.path.dirname(__file__), '../ch02'))
3
4 from program2_1 import Dvector
5 from program2_2 import Dmatrix
6 from program2_3 import input_vector, input_matrix
7 from program2_8 import vector_norm_max
8
9 N = 10 # N元方程式
10 EPS = 10.0 ** -8.0 # epsilon の設定
11 KMAX = 100 # 最大反復回数
12
13 def main():
14     a = Dmatrix(1, N, 1, N) # 行列 a[1...N][1...N]
15     b = Dvector(1, N) # b[1...N]
16     x0 = Dvector(1, N) # x[1...N]
17
18     # ファイルのオープン
19     with open("input_sp.dat", "r") as fin:
20         with open("output_sp.dat", "w") as fout:
21             input_matrix(a, 'A', fin, fout) # 行列 A の入出力
22             input_vector(b, 'b', fin, fout) # ベクトル b の入出力
23             input_vector(x0, 'x0', fin, fout) # 初期ベクトル x0 の入出力
24             x = gauss_seidel(a, b, x0) # ガウス・ザイデル法
25
26             # 結果の出力
27             fout.write("Ax=b の解は次の通りです\n")
28             for i in range(1, N+1):
29                 fout.write("{:.6f}\n".format(x[i]))
30
31 # ガウス・ザイデル法
32 def gauss_seidel(a: Dmatrix, b: Dvector, x0: Dvector, N: int = N):
33     k = 0
34     x = x0.copy()
35
36     x0 = Dvector(1, N) # x0[1...N]
37
38     while True:
39         # x0 <- x_k, x <- x_{k+1}
40         for i in range(1, N+1):
41             x0[i] = x[i] # x_k に x_{k+1} を代入
42         # i=1 の処理
43         x[1] = ( b[1] - sum( ( a[1][j] * x0[j] for j in range(2, N+1) ) ) ) / a[1][1]
44         # i=2,3,...N の処理
45         for i in range(2, N+1):
46             s = sum( ( a[i][j] * x[j] for j in range(1, i) ) ) # i-1列までの和
47             t = sum( ( a[i][j] * x0[j] for j in range(i+1, N+1) ) ) # i+1列以降の和
48             x[i] = ( b[i] - s - t ) / a[i][i]
49         for i in range(1, N+1):
50             x0[i] = x[i]
51         eps = vector_norm_max(x0)
52         k += 1
53
54         if eps <= EPS or k >= KMAX:
55             break
56
57     if k == KMAX:
58         print("答えが見つかりませんでした")
59         exit(1)
60     else:
61         print(f"反復回数は{k}回です")
62         return x
63
64 if __name__ == "__main__":
65     main()

```