

```

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
7
8 def main():
9     print("データの個数を入力してください--->")
10    n = int(input())
11    n -= 1 # データ数がnなので, n <- n-1 として添え字を 0,1,...,n とする
12
13    print("補間点を入力してください--->")
14    xi = float(input())
15
16    x = Dvector(0, n) # x[0...n]
17    y = Dvector(0, n) # y[0...n]
18
19    # ファイルのオープン
20    with open("input_lag.dat", "r") as fin:
21        with open("output_lag.dat", "w") as fout:
22            input_vector( x, 'x', fin, fout ) # ベクトル x の入出力
23            input_vector( y, 'y', fin, fout ) # ベクトル y の入出力
24
25            print("補間の結果は, P({:.6f})={:.6f}".format(xi, newton_ip(x,y,n,xi)))
26
27            # グラフを描くために結果をファイルに出力
28            xi = x[0]
29            while xi <= x[n]:
30                fout.write("{:.6f} \t {:.6f}\n".format(xi, newton_ip(x,y,n,xi)))
31                xi += 0.01
32
33
34    # ニュートン補間
35    # 添字は 0,1,...,n と仮定
36    def newton_ip(x: Dvector, y: Dvector, n: int, xi: float) -> float:
37        pn = 0.0
38
39        a = Dmatrix(0, n, 0, n)
40
41        for i in range(n+1):
42            a[i][0] = y[i]
43
44        # 差商の計算
45        for j in range(1, n+1):
46            for i in range(n-j+1):
47                a[i][j] = ( a[i+1][j-1] - a[i][j-1] ) / ( x[i+j] - x[i] )
48
49        # 補間の計算
50        pn, li = y[0], 1.0
51        for j in range(1, n+1):
52            li *= ( xi - x[j-1] )
53            pn += a[0][j] * li
54
55        return pn
56
57
58 if __name__ == "__main__":
59     main()

```