```python
import os, sys
sys.path.append(os.path.join(os.path.dirname(__file__), '../ch02'))

from program2_1 import Dvector

def main():
    a = 0.0
    b = 3.1415926535897932
    N = 2 # N変数

    y = Dvector(1, N)
    f = Dvector(1, N)

    y[1], y[2] = 1.0, 0.0 # 初期値の設定
    print("分割数を入力してください--->", end="")
    n = int(input())

    rk4_m(y, f, N, a, b, n, FUNC) # ルンゲ・クッタ法


# ルンゲ・クッタ法(N 変数版)
def rk4_m(y: Dvector, f: Dvector, N: int, a: float, b: float, n: int, F):
    k1  = Dvector(1, N)
    k2  = Dvector(1, N)
    k3  = Dvector(1, N)
    k4  = Dvector(1, N)
    tmp = Dvector(1, N)

    # 初期値の設定・表示
    h = (b - a) / n # 刻み幅
    x = a
    print("x={:.6f} \t y1={:6f} \t y2={:.6f} ".format(x, y[1], y[2]))

    # ルンゲ・クッタ法(N 変数版)
    for i in range(n):
        # k1 の計算
        FUNC( x, y, f )
        for j in range(1, N+1):
            k1[j] = f[j]
        for j in range(1, N+1):
            tmp[j] = y[j] + h*k1[j] / 2.0
        # k2 の計算
        FUNC( x+h/2.0, tmp, f )
        for j in range(1, N+1):
            k2[j] = f[j]
        for j in range(1, N+1):
            tmp[j] = y[j] + h*k2[j] / 2.0
        # k3 の計算
        FUNC( x+h/2.0, tmp, f )
        for j in range(1, N+1):
            k3[j] = f[j]
        for j in range(1, N+1):
            tmp[j] = y[j] + h*k3[j]
        # k4 の計算
        FUNC( x+h, tmp, f )
        for j in range(1, N+1):
            k4[j] = f[j]
        for j in range(1, N+1):
            y[j] = y[j] + h/6.0*( k1[j] + 2.0*k2[j] + 2.0*k3[j] + k4[j] )
        x += h
        print("x={:.6f} \t y1={:.6f} \t y2={:.6f} ".format(x, y[1], y[2]))


# 関数の定義
def FUNC(x: float, y: Dvector, f: Dvector):
    f[1] = y[2]
    f[2] = -y[1]


if __name__ == "__main__":
    main()
```