Notes on `Nsite_ring.py`

1. Basic Principle

This homework discusses a **quantum spin-1/2 chain** of N sites arranged in a ring.
The Hamiltonian is given by

$$H = -\sum_{j=1}^{N} \left(g\, \sigma_{j-1}^{z} \sigma_{j}^{y} \sigma_{j+1}^{x} + J\, \sigma_{j}^{z} \sigma_{j+1}^{z} + h\, \sigma_{j}^{x} \right),$$

where $\sigma^{x,y,z}$ are Pauli matrices, while the **periodic boundary condition** shows that $\sigma_{N+1} = \sigma_1$.
This model includes both **two-spin** ($\sigma^z \sigma^z$) and **three-spin** ($\sigma^z \sigma^y \sigma^x$) interactions, with a transverse field h.

The interplay among these competing terms governs the ground-state structure and excitation spectrum.

Because the Hamiltonian is invariant under lattice translation, we can use the **translation operator** T:

$$T|s_1, s_2, \ldots, s_N\rangle = |s_2, \ldots, s_N, s_1\rangle.$$

Its eigenstates satisfy

$$T|\psi_k\rangle = e^{-\mathrm{i}k}|\psi_k\rangle, \qquad k = \frac{2\pi m}{N} \quad (m = 0, 1, \ldots, N-1).$$

The Hilbert space can thus be decomposed into **momentum sectors**, allowing a block-diagonal form of H:

$$H = \bigoplus_k H_k.$$

Then we can calculate the eigenvalues in each block of H. Notice that the lowest eigenvalue across all k gives the **ground-state energy**, and its corresponding eigenvector gives the **ground-state vectors**.

Lastly, the ground-state expectation values are computed as

$$\langle \sigma^x \rangle = \frac{1}{N} \sum_i \langle \psi_0 | \sigma_i^x | \psi_0 \rangle, \quad \langle \sigma^z \rangle = \frac{1}{N} \sum_i \langle \psi_0 | \sigma_i^z | \psi_0 \rangle.$$

2. Source Code of `Nsite_ring.py`

(a) Hamiltonian

Firstly, we preset the parameters and prepare the Pauli matices to represent the spin operaters. Then we define following function:

```
def kron_n(op_list):
    res = np.array([[1]], dtype=complex)
    for op in op_list:
        res = np.kron(res, op)
    return res
```

to get the tensor product of the N-spin system.

For each site j, the local contributions are built as: $-g\,\sigma^z_{j-1}\sigma^y_j\sigma^x_{j+1}$, $-J\,\sigma^z_j\sigma^z_{j+1}$, $-h\,\sigma^x_j$, and summed to form the total H.

(b) Translation Operator

We build the translation matrix T that cyclically shifts basis states by one site, then diagonalizes it to find eigenvalues $e^{-\mathrm{i}k}$:

```
eigvals_T, eigvecs_T = np.linalg.eig(T)
```

(c) Diagonalization in Each Momentum Sector

We initial a dictionary named `momentum_sectors` with keys in 0-N, then we identify the variable `k_index` by following estimation to generate corrsponding keys for `momentum_sectors` :

```
for i, eigenvalue in enumerate(eigvals_T):
    abs_phase = -cmath.phase(eigenvalue)
    if abs_phase < 0:
        abs_phase += 2 * np.pi
    k_index = round(abs_phase * N / (2 * np.pi)) % N
    momentum_sectors[k_index].append(i)
```

For each momentum sector, the Hamiltonian is projected as

$$H_k = V_k^\dagger H V_k$$

where V_k is composed by the eigenvectors of the matrix T:

```
for k_index in range(N):
    indices = momentum_sectors[k_index]
    ...
    Vk = eigvecs_T[:, indices]
    Hk = Vk.conj().T @ H @ Vk
    Ek, psi_k = eigh(Hk)
```

Then we can identify the ground state by:

```
    if Ek[0] < ground_energy:
            ground_energy = Ek[0]
            ground_state = Vk @ psi_k[:, 0]
```

(d) Ground State Properties

$\langle \sigma_x \rangle$ and $\langle \sigma_z \rangle$ can be calculated as follow:

```
for j in range(N):
    ops_x = [id2] * N
    ops_x[j] = sx
    op_x_j = kron_n(ops_x)
    sigmax_exp += np.vdot(ground_state, op_x_j @ ground_state)
sigmax_exp /= N
...
```

In the end, to prominent the output, we build a function `Format_complex` with accuracy at `threshold = 1e-10` , by which we can get the round results.
(see in the code annotation)

3. Numerical Results and Analysis

The output in cases $g = J = h = 1, N = 8$:

```
Case: g=1 J=1 h=1 Ns=8

(1) The eigenvalues in each Momentum_Sector_ki:

ki= 0 : [-11.97450070 -11.89024988 -9.51415508 -9.25511722 -7.85248869 -6.29085969 -5.05463167 -4.04429811 -3.37230998 -2.88661648 -2.8284
2712 -2.44639520 -1.75455059 -1.46410162 -1.40352048 -1.23238656 -0.42483153 -0.00000000 -0.00000000 0.00000000 0.00000000 0.88020197 0.89
940195 2.11883867 2.82842712 2.85399292 3.08806696 3.86534552 4.34331496 4.58045444 5.46410162 6.55887379 7.22620043 8.38209486 11.1301048
2 11.47002058]
ki= 1 : [-9.52059914 -9.14361028 -7.96292754 -6.28512700 -4.56928847 -4.15230775 -3.82710224 -2.76562845 -2.26580240 -1.72080360 -1.362816
96 -1.25373809 -1.09274357 -0.85649737 -0.71798957 0.12866451 0.70294386 1.24417659 1.33391563 2.04239151 2.24284137 2.84133477 3.18876583
 4.93304761 4.95712202 5.34772734 5.36349783 6.68121886 7.16843394 9.32090075]
ki= 2 : [-9.75961101 -8.36870445 -7.13823818 -6.03408247 -5.13717262 -4.87955417 -3.80934503 -3.78807751 -2.94464210 -2.73503675 -2.502571
21 -2.14399472 -1.94598001 -1.69280661 -0.95578852 0.00000000 0.00000000 0.29563433 0.93164293 0.98791904 1.78213904 1.81843430 2.28874153
 2.92187704 3.30266542 3.69366567 4.00000000 4.21675647 5.68436051 6.92987397 7.66588055 8.36862252 8.94739204]
ki= 3 : [-8.98318383 -7.58328975 -6.10427650 -5.27352133 -5.10448089 -4.86633829 -4.35649230 -3.87131019 -3.50959288 -3.28847170 -2.494914
24 -1.49912811 -1.20873263 -0.75921039 -0.70258175 0.41249359 0.75009798 1.08812144 1.82178637 2.27433803 2.50797492 3.21849925 3.51220806
 3.65730447 4.14904570 4.94337970 6.32871281 6.95873375 7.21675557 10.76607314]
ki= 4 : [-7.17412836 -7.16574576 -6.65066490 -6.05348503 -5.98360880 -5.36079645 -4.76029479 -4.72355517 -4.19214598 -3.38933391 -2.688029
77 -2.51255130 -2.26014377 -1.24271202 -1.08426801 -0.09328139 -0.00000000 -0.00000000 0.49863359 1.30534822 2.26014377 2.33195010 2.60344
324 2.71743663 4.00000000 4.52124681 4.63468173 4.76029479 5.30231559 5.35535802 5.66253772 6.65066490 9.60955074 11.12113956]
ki= 5 : [-8.98318383 -7.58328975 -6.10427650 -5.27352133 -5.10448089 -4.86633829 -4.35649230 -3.87131019 -3.50959288 -3.28847170 -2.494914
24 -1.49912811 -1.20873263 -0.75921039 -0.70258175 0.41249359 0.75009798 1.08812144 1.82178637 2.27433803 2.50797492 3.21849925 3.51220806
 3.65730447 4.14904570 4.94337970 6.32871281 6.95873375 7.21675557 10.76607314]
ki= 6 : [-9.75961101 -8.36870445 -7.13823818 -6.03408247 -5.13717262 -4.87955417 -3.80934503 -3.78807751 -2.94464210 -2.73503675 -2.502571
21 -2.14399472 -1.94598001 -1.69280661 -0.95578852 0.00000000 0.00000000 0.29563433 0.93164293 0.98791904 1.78213904 1.81843430 2.28874153
 2.92187704 3.30266542 3.69366567 4.00000000 4.21675647 5.68436051 6.92987397 7.66588055 8.36862252 8.94739204]
ki= 7 : [-9.52059914 -9.14361028 -7.96292754 -6.28512700 -4.56928847 -4.15230775 -3.82710224 -2.76562845 -2.26580240 -1.72080360 -1.362816
96 -1.25373809 -1.09274357 -0.85649737 -0.71798957 0.12866451 0.70294386 1.24417659 1.33391563 2.04239151 2.24284137 2.84133477 3.18876583
 4.93304761 4.95712202 5.34772734 5.36349783 6.68121886 7.16843394 9.32090075]

(2) Ground_State_energy_per_site= -1.496812587859699
(2) Ground_State_sigmax_per_site= (0.4569519539796922 + 1.734723475976807e-18j)
(2) Ground_State_sigmaz_per_site= (1.6316809015037848e-14 + 0.0j)
```

(1) Spectrum Interpretation

The energy dispersion $E(k)$ shows multiple excitation bands.
The smooth variation with k implies quasi-particle-like excitations, similar to magnons in simpler spin models.

(2) Ground-State Properties

- The lowest eigenvalue **-11.97450070** (ground state energy in total) appears at $k = 0$, meaning that the ground state preserves translational symmetry.
- The expectation value $\langle \sigma^x \rangle \approx 0.457$ indicates partial alignment along the transverse field, say the polarization.
- $\langle \sigma^z \rangle \approx 0$ suggests no longitudinal magnetization, showing the field dominates the spin order.
- Imaginary shows no contribution to the expection of observations.

4. Summary

This work implemented a complete diagonalization of a quantum spin chain with periodic boundary conditions.
Through translation symmetry, the computation was simplified by dividing the Hilbert space into momentum sectors.