1. Monte Carlo Methods (Phương pháp Monte Carlo):

- Là phương pháp ước lượng dựa trên việc lấy mẫu ngẫu nhiên lặp đi lặp lại.
- Trong học tăng cường, phương pháp Monte Carlo giúp ước lượng giá trị của hàm giá trị (value function) dựa trên trải nghiệm thực tế mà không cần biết trước xác suất chuyển trạng thái (mô hình môi trường).

2. Value Function (Hàm giá trị):

• Là hàm **ước lượng tổng phần thưởng kỳ vọng** (expected return) từ một trạng thái nhất định, theo một chính sách cho trước.

3. Return (G t):

 Là tổng phần thưởng nhận được từ thời điểm t đến hết episode, có thể có chiết khấu theo hệ số γ (gamma).

•
$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots$$

4. Episodic Task (Nhiệm vụ dạng episode):

 Là nhiệm vụ mà tương tác của agent với môi trường có điểm kết thúc rõ ràng (terminal state), ví dụ như chơi một ván game, hoặc hoàn thành một nhiệm vụ.

5. First-visit vs Every-visit Monte Carlo:

- **First-visit MC**: Chỉ cập nhật giá trị trạng thái tại lần **đầu tiên** nó xuất hiện trong mỗi episode.
- Every-visit MC: Cập nhật giá trị trạng thái **mỗi lần** nó xuất hiện trong episode.

6. Policy (Chính sách):

- Là chiến lược chọn hành động tại mỗi trạng thái.
- Monte Carlo sử dụng chính sách để tạo ra các tập dữ liệu từ tương tác với môi trường.

7. Sample Average Estimation (Ước lượng trung bình mẫu):

- Sau khi quan sát nhiều lần một trạng thái, **trung bình các giá trị return** từ trạng thái đó sẽ tiệm cận giá trị thực của trạng thái.
- Công thức cập nhật giá trị trạng thái S:

$$V(S) = rac{1}{n} \sum_{i=1}^n G_i$$

8. Incremental Update (Cập nhật gia tăng):

 Một cách cập nhật giá trị mà không cần lưu toàn bộ danh sách return, chỉ cần cập nhật theo công thức:

$$V(S) \leftarrow V(S) + \alpha[G - V(S)]$$

Trong đó:

- G là return quan sát được,
- α là learning rate.

9. Ưu điểm của Monte Carlo:

- Không cần biết mô hình môi trường (không cần xác suất chuyển trạng thái như trong Dynamic Programming).
- Dễ triển khai, đặc biệt trong các nhiệm vụ có endpoint rõ ràng (episodic).