|  |  |  |
| --- | --- | --- |
|  | **Recursion** | **Iteration** |
| Core structure | Recursion uses **selection structure**. | Iteration uses **repetition structure**. |
| Infinite Condition | **Infinite recursion**occurs if the **recursion step does not reduce the problem in a manner that converges on some condition** (**base case**) and Infinite recursion can **crash the system.** | An **infinite loop** occurs with iteration if the **loop condition test never becomes false** and Infinite looping uses **CPU cycles repeatedly**. |
| Termination | Recursion **terminates** when a **base case** is recognized. | An iteration **terminates**when the **loop condition fails**. |
| Speed | Recursion is usually **slower than iteration** **due to the overhead of maintaining the stack.** | An iteration does not use the **stack**so it's **faster than recursion**. |
| Memory Usage | Recursion uses **more memory than iteration**. | Iteration consumes **less memory.** |
| Syntax | Recursion makes the **code smaller**. | Iteration makes the **code longer**. |
| Time Complexity | Very high (generally exponential) time complexity. | Relatively lower time complexity (generally polynomial-logarithmic). |
| Preferred | Used when code size needs to be small, and time complexity is not an issue. | Used when time complexity needs to be balanced against an expanded code size. |