Retaining knowledge, especially when it comes to coding and problem solving, is largely about practice, repetition, and understanding. Here's a detailed plan for not forgetting coding solutions and improving problem-solving skills:

Understand, Don't Memorize: It's crucial that you understand the underlying concepts instead of just memorizing the code.

Example: If you're sorting an array, understand why a certain algorithm (e.g., quicksort, mergesort) works and the principle behind it rather than just memorizing the code.

Practice Regularly:

Coding platforms: Websites like LeetCode, HackerRank, and CodeWars offer countless problems to practice coding.

Revisit old problems: After a week or two, try solving problems you've already tackled. It reinforces memory.

Teach Others: Explaining concepts to someone else, even if it's just an imaginary person or a rubber duck, can solidify your understanding.

Diversify Your Problems: Don't stick to one kind of problem. This will give you exposure to different problem-solving techniques and patterns.

Study Theory: While practicing is essential, also invest time in understanding algorithms and data structures theoretically. Books like "Introduction to Algorithms" by Cormen et al. are excellent resources.

Write It Down: Keep a personal notebook or digital note-taking app where you jot down problem solutions, algorithms, and your understanding. Revisit these notes often.

Deep Work: Dedicate uninterrupted time to coding. This means no distractions, just you and the problem at hand. It helps in fostering a deep understanding.

Group Study: Collaborate with peers. Discussing and debating solutions can offer new perspectives.

Rest and Sleep Well: Your brain consolidates memories during sleep. Ensure you're getting enough rest.

Stay Curious: Always be curious about why a certain solution works and why another doesn't. This curiosity will drive you to understand deeper.

Challenge Yourself: Every once in a while, try problems that are slightly above your comfort level.

Implement From Scratch: Occasionally, try writing common algorithms and data structures from memory. For instance, try coding up a linked list or a sorting algorithm without referring to any sources.

Stay Consistent: Like any other skill, consistency is key in programming. Code regularly, even if it's just a small problem or task.

Use C Features: Since you're using C, familiarize yourself with its features. Understand pointers, memory management, structs, etc. deeply.

Post-Mortem Analysis: After solving a problem, analyze what you did right, what you did wrong, and how you can improve next time.

Get Feedback: Share your code with peers or mentors and get feedback. This can provide invaluable insights.

In summary, improving problem-solving ability in code, especially with C or any language, is a blend of understanding the theory, consistent practice, and reflection. Stay patient and persistent, and your skills will certainly grow over time.