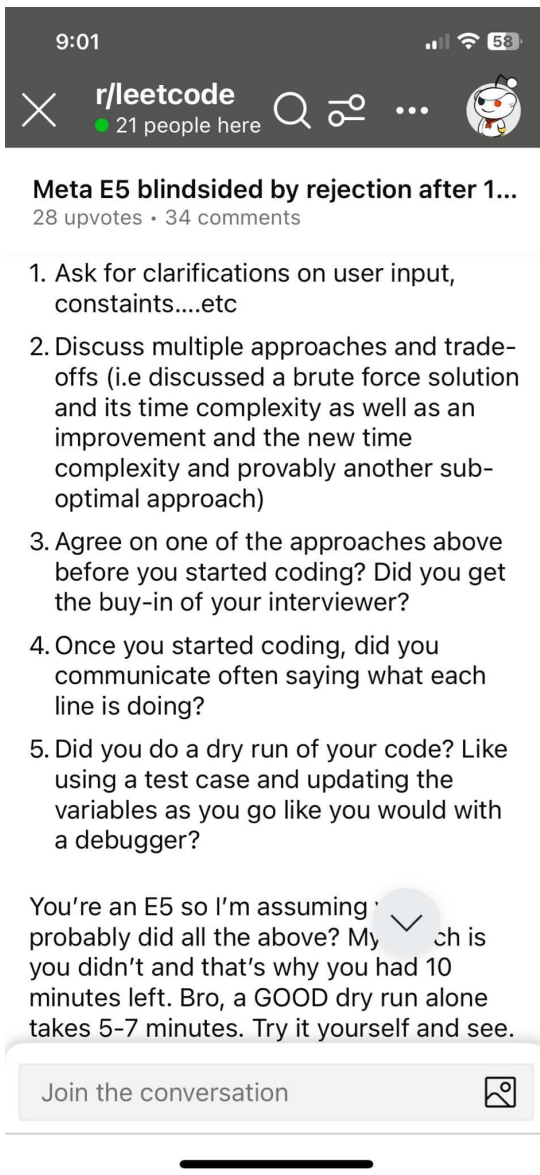


# Template



<https://github.com/mdmzfzl/NeetCode-Solutions>

```
def solve_problem(inputs):  
    # Step 1: Understand the Problem  
    # - Parse inputs and outputs.  
    # - Clarify constraints (e.g., time, space).  
    # - Identify edge cases.  
  
    # Step 2: Plan and Design
```

```
# - Think about the brute-force approach.  
# - Optimize: Can you use dynamic programming, divide & conquer, etc.?  
# - Choose the appropriate data structures (e.g., arrays, hashmaps, heaps).
```

```
# Step 3: Implement the Solution
```

```
# - Use helper functions for modularity.  
# - Write clear, well-commented code.
```

```
def helper_function(args):  
    # Optional: For recursion, BFS, DFS, etc.  
    pass  
  
# Main logic  
result = None # Initialize result or output variable.  
  
# Example logic  
# for num in inputs:  
#     result += num # Or other computations.  
  
return result  
  
# Driver code (for testing locally)  
if __name__ == "__main__":  
    inputs = [] # Replace with example test cases.  
    print(solve_problem(inputs))
```

---

Revision #3

Created 28 December 2024 00:14:27 by victor

Updated 5 April 2025 22:50:32 by victor