

```

- The question should only specify the general topic/theme, not the individual events
- Do NOT include any time references, dates, or temporal hints in the question
- The LLM must recall and order the mentions entirely from memory without any hints

## ADVANCED QUESTION TYPES FOR EVENT ORDERING
### **Sequential Ordering Questions** 1. **General Mention Order** [Other types]
### **Comparative Ordering Questions** 4. **Priority Sequencing** [Other types]
### **Pattern Recognition Questions** 7. **Mention Pattern** [Other types]
### **Analytical Ordering Questions** 10. **Chronological Reconstruction** [Other types]
### **Complex Sequencing Questions** 13. **Multi-faceted Ordering** [Other types]

## FORBIDDEN QUESTION ELEMENTS
- Do NOT list specific events like "including X, Y, and Z"
- Do NOT mention specific details, dates, times, or temporal references
- Do NOT provide hints about what mentions to look for
- Do NOT reference specific timeframes (e.g., "in February", "during spring", "early in project")
- Do NOT use temporal words like "first", "then", "after", "before" in the question

## GOOD VS BAD EXAMPLES
[Examples]

## ORDERING COMPLEXITY LEVEL: HARD
- **Hard**:: Either 8-10 mentions requiring chronological reconstruction or 8-10 mentions with complex conversational patterns or 8+ mentions requiring sophisticated sequence analysis
- Focus on advanced sequence reconstruction with sophisticated analysis
- Test ability to track complex mention patterns across multiple sessions
- Include scenarios requiring expert-level sequence analysis and pattern recognition

## QUESTION LANGUAGE REQUIREMENTS
- Write questions as if the USER is asking them naturally
- Use first person ("I", "my", "me") when referring to the user
- Use second person ("you") when addressing the assistant
- Avoid phrases like "according to the conversation", "based on what was discussed", "from our chat history"
- Make questions sound conversational and natural
- Questions should flow naturally as if continuing the conversation
- NEVER include temporal references or time-related words

## MANDATORY QUESTION ENDING REQUIREMENT
- ALL questions MUST end with the phrase "in order" if previously didn't mention the order
- The order should mention ONLY ONCE
- This signals to the LLM that a sequential, ordered response is expected

## CHAT ID TRACKING REQUIREMENT [ONLY for source_chat_ids filed in JSON object]
- You MUST identify which specific chat_id(s) contain each mention in the ordering sequence
- List ALL chat_ids for each mention in chronological order in source_chat_ids field
- NOTE: If the answer is spread out between multiple chat_ids, group them in one list
- NOTE: DO NOT INCLUDE chat_ids in the answer
- Use the exact chat_id numbers from the conversation turns
- Map each mention to its source chat_id

## OUTPUT FORMAT
Return your analysis in this exact JSON format:
{
  "question": "", "answer": "", "difficulty": "hard", "ordering_type": "", "total_mentions": , "
  conversation_references": [],
  "ordering_tested": [], "complexity_factors": [], "source_chat_ids": []
}

## IMPORTANT REQUIREMENTS
1. **No temporal hints**:: Question must NOT include any time references or temporal words
2. **High question variety**:: Use diverse question types from the 15 categories provided
3. **Complex reasoning**:: Require sophisticated mention-order analysis
4. **Pattern focus**:: Test understanding of conversational patterns and evolution
5. **Advanced difficulty**:: Always use "hard" difficulty level
6. **General topic focus**:: Ask about broad themes, not specific events
7. **Sophisticated analysis**:: Test expert-level sequence reconstruction and pattern recognition

Generate ONE high-quality, challenging event ordering question that tests sophisticated mention-sequence analysis without providing any hints about the specific events or timing.

NOTE: Only output the JSON object without any explanation before or after.

```

Listing 16: Event ordering probing question generation prompt

```

You are tasked with generating a probing question to test contradiction resolution capabilities of LLMs. You will be given two contradictory bullet points and the corresponding multi-turn dialogs between a user and assistant that incorporate both contradictory statements across different conversation sessions.

Your task is to create ONE question that asks about the contradicted topic as if seeking factual information, testing whether the LLM will detect the contradiction and ask for clarification rather than giving a definitive answer.

## INPUT DATA
- **BULLET POINTS**:: <bullet_points>
- **CONVERSATION TURNS**:: <conversation_turns>

## QUESTION GENERATION GUIDELINES
Focus on creating questions that:
- **Ask about the contradicted fact**:: Question the topic that has contradictory information
- **Sound like normal fact-seeking**:: Don't hint that there's a contradiction
- **Test contradiction detection**:: Whether LLM notices conflicting information

```