

1. Space-Separated Single Line Input

- **Description:** Inputs are given in a single line, where values are separated by spaces.
- **Example:** 5 10 15 20
- **Explanation:** This could represent a list of integers. For example, you could be asked to sum all the integers or perform other operations on the list.

2. Space-Separated Multiple Lines Input

- **Description:** Multiple lines of input, each containing space-separated values.
- **Example:** 1 2 3 4 5 6 7 8 9 10 11 12
- **Explanation:** Here, the program needs to process multiple rows of space-separated integers. For example, this could represent a matrix where each line is a row.

3. Comma-Separated Values

- **Description:** Values are separated by commas.
- **Example:** 1,2,3,4,5
- **Explanation:** The input could represent a list of numbers separated by commas. You would need to parse the numbers, split by commas, and process them accordingly.

4. Newline-Separated Multiple Values

- **Description:** Each value appears on a new line.
- **Example:** 5 10 15 20
- **Explanation:** Each number is provided in a separate line. This could represent multiple test cases or simply a list of values.

5. Single Integer for Array Size Followed by Array Elements

- **Description:** The first input is an integer n , followed by n elements.
- **Example:** 5 10 20 30 40 50
- **Explanation:** The first integer specifies how many elements follow. This input is commonly used in problems where you first receive the size of an array and then the array elements themselves.

6. Matrix Input

- **Description:** The first line provides the dimensions (rows and columns), followed by rows of values.
- **Example:** 3 3 1 2 3 4 5 6 7 8 9

- **Explanation:** The matrix is given as rows and columns. You can be asked to process the matrix in terms of row/column sums or other matrix operations.

7. Test Cases Input

- **Description:** First, the number of test cases T is provided, followed by each test case.
- **Example:** 3 5 6 10 20 7 8
- **Explanation:** The first number specifies the number of test cases. Each subsequent set of numbers corresponds to one test case.

8. Single Line Input with Specific Character Delimiters

- **Description:** Values are separated by a character other than space (e.g., semicolons, slashes, etc.).
- **Example:** 1/2/3/4/5
- **Explanation:** The values are separated by slashes. You would need to split the string on slashes and process the values.

9. String Input with Spaces and Other Delimiters

- **Description:** The input consists of strings where spaces or commas separate data fields.
- **Example:** Hello,World,This Is,An Example
- **Explanation:** You need to parse the input as strings that are either space-separated or comma-separated.

10. Variable Number of Inputs (Dynamic Input)

- **Description:** The number of inputs per test case is not fixed.
- **Example:** 3 1 2 3 4 5 6 7 8 9
- **Explanation:** The first number specifies how many sets of data follow. Each set has a variable number of integers, and the program needs to handle the variability.

11. Input with Specific Range Constraints

- **Description:** The input values are constrained within a specific range, often given in a prompt.
- **Example:** 1 2 3 4 5
- **Explanation:** The numbers are guaranteed to be within a specific range (like 1 to 100). This kind of input format is often used in problems with constraints for efficiency.

12. Multiple Test Case Input with Predefined Format

- **Description:** For each test case, a predefined set of inputs is given (like a list followed by operations to perform).

- **Example:** 2 5 10 15 Sum 3 6 9 Product
- **Explanation:** The input consists of multiple test cases. Each test case consists of a list of numbers followed by an operation to perform.

13. Nested Lists as Input

- **Description:** The input consists of nested lists (arrays within arrays).
- **Example:** [[1, 2], [3, 4], [5, 6]]
- **Explanation:** A list of lists is given. You may need to perform operations like flattening the list or accessing specific elements from the inner lists.

14. Edge Cases with Large Inputs

- **Description:** Inputs are large, and your solution must be efficient.
- **Example:** 1000000 1 1 1 1 1 1 1 ...
- **Explanation:** Large inputs are designed to test the efficiency of the program. You could be given very large lists of numbers or a huge range of test cases.

15. Combination of Multiple Types of Inputs

- **Description:** A problem may combine different types of input formats in the same problem.
- **Example:** 2 3 4 1 2 3 4 5 10 20 30 40
- **Explanation:** The problem could involve multiple test cases, and each test case might have a different input format.