Here’s a structured template, using the problem-solving methodology, tailored specifically for you as an individual with autism, ADHD, and Asperger's. This template will help you approach complex tasks systematically, breaking them into manageable parts and addressing your unique cognitive processing needs.

Problem-Solving Template for Neurodivergent Individuals

1. Problem Definition:

- **Task:** Clearly state the problem or task you're working on.
- **Example:** *Developing a data compression algorithm using machine learning techniques.*

2. Break Down the Problem (Sub-problems):

- **Step 1:** Identify smaller sub-tasks that make up the larger problem.
  - **Sub-task 1:** Research machine learning models for data compression.
  - **Sub-task 2:** Design the algorithm’s structure (input-output, learning steps).
  - **Sub-task 3:** Train the algorithm on sample datasets.
  - **Sub-task 4:** Test the algorithm’s performance and refine it.

3. Implement SMART Goals for Each Sub-task:

- **Specific:** What exactly will you achieve?
- **Measurable:** How will you track progress?
- **Achievable:** Ensure the task is reasonable based on your time and resources.
- **Relevant:** How does this task connect to the overall problem?
- **Time-bound:** Set a deadline for each task.

Example for Sub-task 1:

- **Specific:** Complete research on the top 3 machine learning models for compression.
- **Measurable:** Write a 2-page summary of each model.
- **Achievable:** Dedicate 2 hours each day for research.
- **Relevant:** Helps inform which model to use for your algorithm.
- **Time-bound:** Finish research within 1 week.

4. Apply Sensory Considerations & Structured Environment:

- **Quiet Workspace:** Set up a distraction-free workspace with minimal sensory input.
- **Visual Aids:** Use charts, diagrams, or visual timelines to help organize your thoughts.
- **Scheduled Breaks:** Schedule short breaks (e.g., every 30 minutes) to prevent overwhelm.

Example:

- Use a timer to remind you to take a 5-minute break after every 30 minutes of focused work.
- Include visuals like flowcharts to map out how your algorithm compresses data.

5. Graph Structure for Problem Navigation:

- **Visual Representation:** Use a graph or mind map to break down the complexity of the project.
- **Difficulty Levels:** Create layers where easier tasks are at the top, and more challenging ones are at the bottom.

Example:

- Top Layer: Research machine learning models (easier).
- Middle Layer: Implement model and train algorithm (moderate).
- Bottom Layer: Fine-tune performance and troubleshoot (more difficult).

6. PARA Structure for Organization:

- **Projects:** Break the overall project into major categories (Research, Development, Testing).
- **Areas:** Focus on areas like data collection, algorithm design, or model training.
- **Resources:** Create folders or sections to organize all the research papers, code snippets, and documentation.
- **Archives:** Archive old versions of the algorithm, test results, or research notes for reference.

Example PARA Setup:

- **Projects:** "Data Compression Algorithm Development."
- **Areas:** “Algorithm Design,” “Dataset Training.”
- **Resources:** Folder with research papers on compression models.
- **Archives:** Old versions of code and project notes.

7. Scheduled Reviews & Flexibility:

- **Weekly Review:** Set a time each week to review progress, update SMART goals, and adjust tasks based on what’s working or not.
- **Flexible Adjustments:** Adjust deadlines or strategies if needed to reduce stress, ensuring you stay engaged without feeling overwhelmed.

Example Review:

- At the end of each week, review if sub-tasks were completed as planned.
- If something took longer than expected, adjust the timeline for the next week.

8. Conclusive Documentation:

- **Document Findings:** Write summaries of what you’ve accomplished and any adjustments you made to the process. This can help you stay

organized and reflect on your progress.

Example:

- After completing research on compression models, document the pros and cons of each model to help you decide which one to implement.

Outcome:

This structured approach will guide you through complex tasks by breaking them into smaller, achievable steps while accommodating your unique needs. By following this template, you can reduce overwhelm, maintain clarity, and stay organized as you work on your data compression algorithm or any other project.

Would you like to modify any of these steps to better suit your workflow or cognitive style?