Test Plan Document

Introduction

The purpose of this test plan is to outline the organization and execution of tests for the command-line application. The scope includes validating the command processing based on a set of test cases stored in specific directories. The primary objective is to ensure the correct functionality of the application under various inputs and scenarios.

Test Organization

Directory Structure:
All test plans are stored in the central directory TestCase.
Subdirectories for each test case command are named

Test Case Structure:
Each test case directory contains:

input.txt: Test input file for the corresponding iteration.
output.txt: Expected output file for the corresponding iteration.
The first number describes which input or output turn would the input or output be and the second number shows which iteration of the said input and output it is.

Test Execution

Script Type:
A DOS Batch File named run_test.bat.

Execution Flow:
The batch file prompts the user for the directory name.
Validates the input; if invalid, prompts for re-entry.
Look for input.txt and output.txt in the specified directory.
Runs the command using the input file.
Compares the output with the expected output.
Reports pass/fail for the current iteration.
Iteration Handling:
Follows the numerical order of input and output files.
Invalid inputs and outputs come before their respective valid counterparts.

Command Structure:

Example command: run_test.bat <directory_name>

Output Handling

Logging:
Log test results in a text file named test_log.txt.
Logs include directory name, iteration number, and pass/fail status.

Reporting

Comparison of output and expected output stored in log files.
Differences are highlighted in the log files.
Summary report provided at the end of the test execution.

Future Considerations

Discussion on Batch File vs shell scripts:

Consider and discuss using a DOS batch file for scalability and maintainability.

Conclusion

This test plan outlines the structure and execution of tests for the command-line application. It provides a clear framework for validating the application's functionality through a set of organized test cases.