Software Testing

Assignment

Software Testing Assignment No: 1

Submitted By:

Muhammad Bilal	22F-3845
Muhammad Ahmed	22F-8814

Submitted To:

Sir Usman Ghous

February 2026

Contents

1 Introduction

This assignment presents a comprehensive analysis of three distinct features through control flow graph construction, cyclomatic complexity calculation, and test path identification using set theory. The features analyzed include Pagination Logic (dividing file content into fixed-size pages), Search & Replace Word (keyword searching with prefix extraction), and Auto-Save Logic (automatic file saving with word count validation).

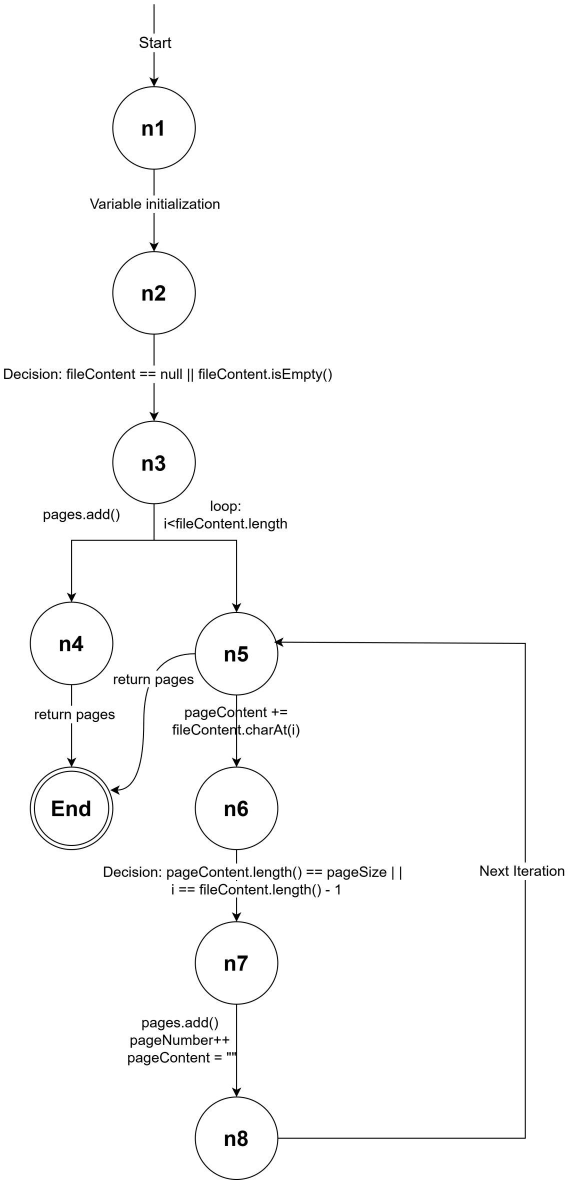
2 Pagination Logic Feature

2.1 Control Flow Graph (CFG)

2.1.1 Step 1: Identify Basic Blocks and Decision Points

Node	Description (Basic Block)
N1	Method entry and initialization: pageSize=100, pageNumber=1, pageContent="", pages = new ArrayList()
N2	Decision: if (fileContent == null || fileContent.isEmpty())
N3	True branch: pages.add(new Pages(...)); return pages;
N4	Loop header: initialize i=0; decision i < fileContent.length()
N5	pageContent += fileContent.charAt(i)
N6	Decision: if (pageContent.length() == pageSize || i == fileContent.length() - 1)
N7	pages.add(new Pages(...)); pageNumber++; pageContent = ""
N8	Loop increment: i++ and back to loop condition (N4)
N9	Method exit: return pages (after loop completion)

2.1.2 Step 2: CFG Diagram



2.1.3 Step 3: Edge List

Edge	From → To	Condition
e1	N1 → N2	Method entry to null/empty check
e2	N2 → N3	fileContent == null || fileContent.isEmpty() is true
e3	N2 → N4	fileContent == null || fileContent.isEmpty() is false
e4	N4 → N5	Loop condition i < fileContent.length() is true
e5	N4 → N9	Loop condition i < fileContent.length() is false (loop exit)
e6	N5 → N6	After appending character to pageContent
e7	N6 → N7	pageContent.length() == pageSize || i == fileContent.length() - 1 is true
e8	N6 → N8	pageContent.length() == pageSize || i == fileContent.length() - 1 is false
e9	N7 → N8	After adding page and resetting pageContent
e10	N8 → N4	Loop increment and next iteration (i++)

2.2 Cyclomatic Complexity Calculation

Using the formula:

$$V(G) = E - N + 2P \tag{1}$$

Where:

- E (Edges) = 10
- N (Nodes) = 9
- P (Connected components) = 1

$$V(G) = 10 - 9 + 2(1) = 3 \tag{2}$$

Verification using decision count:
Decision nodes in the CFG:

- N2: null or empty check
- N4: loop condition
- N6: page size or last character check

Although three predicate nodes appear in the source code, the loop structure is represented as a single cycle in the CFG. Therefore, the number of **linearly independent paths** for this graph is:

$$\boxed{V(G) = 3}$$

This means that at least **three independent test paths** are required for complete basis path coverage.

2.3 Test Paths (Set Theory)

2.3.1 Independent Path Set

The set of independent paths through the CFG is defined as:

$$P = \{p_1, p_2, p_3\} \tag{3}$$

Path 1 (Empty input):

$$p_1 = \langle N_1, N_2, N_3 \rangle$$

Path 2 (Single short page):

$$p_2 = \langle N_1, N_2, N_4, N_5, N_6, N_7, N_8, N_4, N_9 \rangle$$

Path 3 (Multiple pages):

$$p_3 = \langle N_1, N_2, N_4, N_5, N_6, N_8, N_4, N_5, N_6, N_7, N_8, N_4, N_9 \rangle$$