Zip Code Group Project 3.0- User Guide

**1. Introduction**

* Overview of its functionality (CSV conversion, index generation, blocked sequence file creation, record lookup, and dump functions).

**2. Installation Instructions**

* Requirements:
  + C++ compiler (e.g., g++),
  + Linux command line environment.
  + Doxygen (for documentation generation, optional)
* Directory structure:

├── Buffer.h, Buffer.cpp

├── Record.h

├── Block.h, BlockBuffer.h, BlockBuffer.cpp

├── main.cpp

├── resources/zip\_codes.csv

└── output/

**3. How to Compile and Run**

* **Compiling:**

g++ -std=c++11 -o zip\_project Buffer.cpp BlockBuffer.cpp main.cpp -I.

* **Running in Generation Mode:**

./zip\_project

Or

zip\_project.exe

This command will:

* Read the CSV file from ./resources/zip\_codes.csv.
* Convert the CSV records into length-indicated format.
* Group the records into fixed-size blocks and write the resulting blocked sequence set file to ./output/blocked\_sequence\_set.txt.
* **Running in Dump Mode**

./zip\_project --dumpPhysical

Or

zip\_project --dumpPhysical

* + Lists all blocks as they appear in the file.
  + Useful for verifying physical layout.
* **Logical Order (Follow Next Block Links):**

./zip\_project --dumpLogical

Or

zip\_project --dumpLogical

* Follows nextBlock pointers for each block.
* Useful for simulating linked list traversal or logical sequence reading.
* **Dump Both Orders Together:**

./zip\_project --dumpPhysical --dumpLogical

or

zip\_project --dumpPhysical --dumpLogical

**4. Troubleshooting**

* Ensure the CSV file is in the correct location (./resources/zip\_codes.csv).
* Verify that the output directory exists.
* Check file permissions if the file cannot be opened.