Blackbox Testing

The blackbox testing of the program aims to ensure that the program is able to fufil its functions as specified in its specifications. The use case diagram will be referenced for the design of the tests.

The entire program will be treated as a blackbox. Only the input to the program will be considered for equivalence class partitioning.

The program supports only CSV files with the same columns in both files

Equivalence Class Partitioning

1. Characteristics will first be obtained from the input domain of the program, via interface-based input domain modelling and the specifications of the program.
2. The characteristics will then be partitioned.

Input characteristics

1. files' name
2. files' extension
3. files' headers
4. files' header order
5. column number format

Partitions

Specification	Valid Partition	Invalid Partition	Rationale
File name exists in the same directory as the program	True	False	the program requires that the files to be compared be in the same directory as the program

Specification	Valid Partition	Invalid Partition	Rationale
File extension == .csv	True	False	the program only works with CSV files, checking if the file has a .csv extension will be sufficient in checking if the program should be able to run
file1 headers == file2 headers	True	False	the program requires both csv files to contain the same headers for the comparison
file1 header order != file2 header order	Results matches expectations	Results are different from expectations	the program should work regardless of the order of the headers
column numbers separated by commas	True	False	the program should not work if >1 column number is specified and the column numbers are not separated by commas

Boundary Value Analysis

Characteristic	Boundary Value just within the boundary (Valid Partition)	Boundary Value just outside the boundary (Invalid Partition)	Rationale
File name exists in the same directory as the program	sample.csv and sample.csv present the same directory as the program	sample.csv and sample.csv not present the same directory as the program	The program should work as long as it is able to open the file specified via the input, the boundary is whether the file name exists in the directory where the program resides

Characteristic	Boundary Value just within the boundary (Valid Partition)	Boundary Value just outside the boundary (Invalid Partition)	Rationale
File extension == .csv	sample.csv	sample.tsv	the program only works with CSV files, checking if the file has a .csv extension will be sufficient in checking if the program should be able to run
file1 headers == file2 headers	Input 2 files: header of file1 = [a,b,c], header of file2 = [a,b,c]	Input 2 files: header of file1 = [a,b,c], header of file2 = [a,a,b]	the boundary values checks if the program can execute even if the headers of file 2 is a subset of file1's headers
file1 header order != file2 header order	Input 2 files with the same set of column headers, regardless of order: header of file1 = [a,b,c], header of file2 = [c,a,b] and header of file1 = [a,b,c], header of file2 = [c,a,b]	Input 2 files with different sets of column headers: header of file1 = [a,b,c], header of file2 = [c,a,b]	The boundary values check if the program is able to work correctly when given 2 files with headers in the same order or in a different order.
column numbers separated by commas	1,2,3 or 1, 2, 3	1 2 3	the boundary values check if the program still works if the column numbers are separated by delimiters other than commas