A shipping system routes items by code which as follows

Start with a three letter prefix CHI or DET indicating Chicago or Detroit. These are the only two destinations, these are handled differently by the system.

1. Three letters long
2. Upper or lower case
3. DET
4. CHI
5. Two letters long
6. Four letters long
7. Three but not all letters
8. Thee letter code other than DET or CHI

The fourth position is either a * or /, they are treated the same. This is to accommodate legacy codes.

1. Character is * or /
2. Character not * or /

The fifth position is a product category indicated by a letter from ASCII A-T and all processed the same

1. {A B ... T, a, b...t} Any letter on the list
2. Any letter not on the list

The last three position are either 1, 2 or 3 numeric digits long represent a sales code and all processed the same

1. exactly one
2. exactly two
3. exactly three
4. They are digits
5. no digits
6. four digits
7. embedded blanks

8. nondigit

All codes must 6- 8 characters long

1. 6-8
2. Too many
3. Too few

If the code is invalid and error should be printed in the log and the code ignore.

The code should not be case sensitive chi = CHI

1. Critique the spec.
2. Develop a set of equivalence class for test input
   • Listing the valid criteria
3. Choose test cases using boundary value analysis.
   • Choose the valid test cases first
   • Break each test case to get an invalid

Test Cases

DET*a879

chi/T7

DE*a879

chic/T7

D3T*a879

DAL*t89

DET\879

DET*a8 9

DET*a8X9

chi/A

DET*a8789

no input

“The Similarity Principle.”

1. Make the valid cases as dissimilar as possible

2. Make each invalid test case exactly like a valid case but differing in one criteria