**DIT636 / DAT560 - Test Design Activity**

You have created a service intended to find all instances of a **pattern** in a **file**.

You want to now design test specifications using the process discussed in Lectures 5-6.

find(pattern,file)

This pattern can contain spaces and quotes. For example:

find(“john”,myFile)

Finds all instances of john in the file

find(“john smith”,myFile)

Finds all instances of john smith in the file

find(““john” smith”,myFile)

Finds all instances of “john” smith in the file

Identify valid test specifications for this function.

1. **Identify the choices that you control when testing.**
   * For each input parameter (pattern, file), what aspects should you consider when testing?
   * Are there any additional environmental factors that you can control that will also impact the outcome of executing this service?
2. **For each choice, identify representative values that could lead to different outcomes of executing this service.**
   * These are the options you can select for that choice.
   * Try to group related values that lead to the same outcome (e.g., “positive integer” instead of 1, 2, 3, 4, 5, …)
3. **Impose constraints on the representative values of different choices to reduce the pool of test cases.**
   * **error** constraints identify representative values that should result in an error no matter what values are selected for the *other* choices.
   * **single** constraints identify representative values that should result in normal execution, but should be tried once because they have the potential for error or strange behavior.
   * **if-constraints** identify representative values for choices that should only be used if particular values are used for other choices.
     + (“if choice X = THIS, then choice Y = THAT”)