Annie Steenson
SE 417

Part 1:

Use that CFG to create a test set which achieves the most complete edge-pair coverage you can find for the getParametersFromFile() method;

path(t1) = {1, 2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20}
path(t2) = {1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20}
path(t3) = {1, 13, 14, 15, 20}

difference between t1 and t2 are nodes 6 and 7 respectively.
node 6 represents if cols > 1
node 7 represents if cols <= 1

Need to test both (cols > 1 and cols <= 1)

T = {t1, t2, t3} provides most complete edge-pair coverage

Define an uni-dimensional input partitioning:
Uni-dimensional input partitioning: Consider one input variable at a time.

Parameter Placeholder for the name = filename

Partition Placeholder Name	w	x	y
Partition Specification	Valid Filename	Valid Filename, empty file	Invalid Filename
Expected Return Value	params[num of rows][cols]	null	null

Parameter Placeholder for the age = cols

Partition Placeholder Name	1	2
Partition Specification	cols = number of columns in file	cols != number of columns in file
Expected Return Value	params[num of rows][cols]	null

Part 2:

Parameter Placeholder for A = String word

Partition Placeholder Name	w	x	y	z	a
Partition Specification	Empty String	String with (uppercase) O	String with multiple (lowercase) o	String with tab character	null
Expected Return Value	0	number of O's present	number of o's present	IllegalInputException	NullPointerException

b) Give an All Combinations test set for the partitioning:

All Combinations

w	x	y	z	a

Test Set:
{ (w), (x), (y), (z), (a) }