Kristopher Ables

Dr Mehra Borazjany

6359.001

November 19, 2022

1) The function takes the average of the integers in a list starting from the first index to the kth index (or the average of the entire list if k is greater than the list length)
2)

#	Description	Input	Expected Output
1	List is empty	K can be any integer, list is empty	0
2	K is equal to list length	K is equal to the length of a list of any length greater than 0	The total average of a list
3	K is less than list length	K is less than the length of a list of any length greater than 0	The average of the list up to index K
4	K is <=0	K is a value less than or equal to 0, and the list is not empty	0

#	Description	Input	EO	AO	Result
1	Test for Empty List	K=5, List = {}	0	0	True
2	Test for a K equal to list length	K = 10, List = {-11, 34, 26, -26, 50, 4, -11, 47, -11, 38}	14	14	True
3	Test for K less than list length	K = 1, List = {-11, 34, 26, -26, 50, 4, -11, 47, -11, 38}	-11	-11	True
4	Test for K less than or equal to 0	K = -5, List = {-11, 34, 26, -26, 50, 4, -11, 47, -11, 38}	0	0	True

3) The input domain is a complete set of integers and a complete set of all possible integer lists. This may be partitioned as follows:

   Partition 1: K <=0 or list.length = 0

   Partition 2: K < list.length

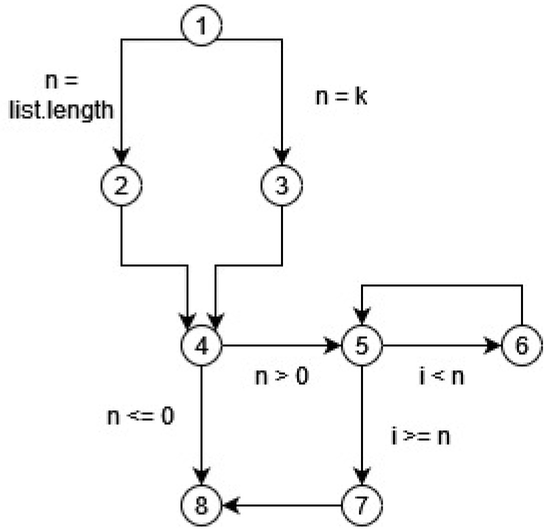
   Partition 3: K >= list.length

4) The boundary tests are as follows:

K	List
0	Empty List
-1	List of 1 element
1	List of 2 elements
List.length	List with the upper bound of elements
List.length - 1	List with the upper bound of elements - 1
List.length+1	List with the upper bound of elements +1
MAX_INTEGER	
MIN_INTEGER	

5) Due to memory constraints, an array consisting of the MAX_INTEGER is not possible. Therefore, the upper bounds of the list were set to an arbitrary very large number (1,000,000 elements). Otherwise, all tests were successful and there were no problems detected with the initial code, so long as the expected outcome is an integer truncated of its decimal values.

6) A tool was not utilized to calculate branch effectiveness, however in calculating accessed branches manually we can see that all branches were accessed



list = {-11, 34, 26, -26, 50, 4, -11, 47, -11, 38} and k = 15 covers {1,2,4,5,6,7,8}
list = {-11, 34, 26, -26, 50, 4, -11, 47, -11, 38} and k = -5 covers {1,3,4,8}
list = {-11, 34, 26, -26, 50, 4, -11, 47, -11, 38} and k = 5 covers {1,3,4,5,6,7,8}
list = {-11, 34, 26, -26, 50, 4, -11, 47, -11, 38} and k =1 covers {1,3,4,5,7,8}
list = {} and k =5 covers {1,2,4,8}
list = {1} and k =5 covers {1,3,4,5,7,8}