Test Plan Homework 1

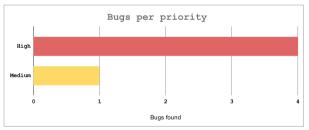
Tester names: Nicola Balzano, Pasquale
Di Gennaro

Environment used: Intellij IDEA 2023.2.3

Project: onedrive

ID	PRIORITY	TESTER NAME	STEP	DESCRIPTION	listLeftOfStrings	INPUT listRightOfStrings	STATUS	EXPECTED RESULT	ACTUAL RESULT	BUG FOUND?	BUG DESCRIPTION	SOLVED	COMMENT
1	er d sala	December 1	1	listofDisktChrises aull	"sinistra",			11	null				
2	High	Pasquale	1	listOfRightStrings null	"sinistra sinistra"	null	passed	null	nuii				
3	High	Nicola	1	listOfLeftStrings null	null	"destra", "destra destra"	passed	null	null				
4	High	Pasquale	1	stringa in listOfLeftStrings vuota	"", "sinistra"	"destra", "destra"	passed	"destra", "sinistradestra"	"destra", "sinistradestra"				
4	High	Nicola	1	stringa in listOfRightStrings vuota	"sinistra", "sinistra"	"destra", ""	passed	"sinistradestra", "sinistra"	"sinistradestra", "sinistra"				
5					"sinistra",								
6	High	Pasquale	1	listOfRightStrings vuota	"sinistra"	empty	passed	"sinistra", "sinistra"	"sinistra", "sinistra"				
	High	Nicola	1	listOfLeftStrings vuota	empty	"destra", "destra"	passed	"destra", "destra"	"destra", "destra"				
7	High	Pasquale	1	liste di lunghezza 1	"sinistra"	"destra"	passed	"sinistradestra"	"sinistradestra"				
8											il programma non inserisce lo		
	High	Nicola	1	listOfLeftStrings contenente numero che segue un carattere e contenente un numero che precede un carattere	"sinistral", "lsinistra"	"destra", "destra"	failed	"sinistra 1destra", "1 sinistradestra"	"sinistra ldestra", "lsinistradestra"	~	spazio per dividere un numero da un carattere adiacente se quest'ultimo segue il numero.	~	
9				listOfRightStrings contenente									
	High	Pasquale	1	stringhe rispettivamente con i caratteri "no" e " "	"sinistra", "sinistra"	"destrano", "des tra"	passed	"sinistradestra", "sinistradestra"	"sinistradestra", "sinistradestra"				
10	High	Nicola	1	length(listOfLeftStrings) <length< th=""><th>"sinistra"</th><th>"destra", "destra"</th><th>naccad</th><th>"sinistradestra", "destra"</th><th>"sinistradestra", "destra"</th><th></th><th></th><th></th><th></th></length<>	"sinistra"	"destra", "destra"	naccad	"sinistradestra", "destra"	"sinistradestra", "destra"				
11	nign	NICOIA		(listofRightStrings)	SINISCIA	destra , destra	passed	simistradestra , destra	Simistradestra , destra				
	High	Pasquale	1	<pre>length(listOfLeftStrings)>length (listofRightStrings)</pre>	"sinistra", "sinistra"	"destra"	passed	"sinistradestra", "sinistra"	"sinistradestra", "sinistra"				
12	High	Nicola	1	length(listOfLeftStrings)+length	"sinistra"	"destradestra"	nassed	"sinistradestradestra"	"sinistradestradestra"				
13				(listOfLeftStrings)=20			,						
	High	Pasquale	1	<pre>length(listOfLeftStrings)+length (listOfLeftStrings)=21</pre>	"sinistra "	"destradestra"	passed	sinistraDestradestra	sinistraDestradestra				
14	High	Nicola	1	length(listOfLeftString)=14	"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14"	"destra"	passed	"ldestra", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14"	"ldestra", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14"				
15	High	Pasquale	1	length(listOfRightString)=14	"sinistra"	"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14"	passed	"sinistral", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14"	"sinistra1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14"				
16	High	Nicola	1	length(listOfLeftString)=15	"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15"	"destra"	passed	"1", "2", "3", "4", "5", "6", "7", "8", "10", "11", "12", "13", "14", "15"	"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15"				
17													
	High	Pasquale	1	length(listOfRightString)=15	"sinistra"	"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15"	passed	"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15"	"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15"				

18													
	High	Nicola	1	listOfLeftStrings contenente stringhe rispettivamente con "-1", " 1-" e "-1-"	"sinistra-1", "sinistra 1-", "sinistra-1-"	"destra", "destra", "destra"	failed	"sinistra- ldestra", "sinistra 1 -destra", "sinistra- 1 -destra"	"sinistra-ldestra", "sinistra l-destra", "sinistra-l-destra"	✓	I caratteri speciali non vengono considerati dal metodo, quindi tutte in tutte le stringhe contenenti caratteri speciali adiacenti ad un numero il carattere "[space]" non viene inseri	V	
19													
	High	Pasquale	1	<pre>listOfLeftStrings contenente stringhe rispettivamente con "!1", " 1!" e "!1!"</pre>	"sinistra!1", "sinistra 1!", "sinistra!1!"	"destra", "destra", "destra"	failed	"sinistra! ldestra", "sinistra l !destra", "sinistra! l !destra"	"sinistra!ldestra", "sinistra 1!destra", "sinistra!l!destra"	~	-	~	
20													
	High	Nicola	1	listOfLeftStrings contenente stringhe rispettivamente con "@1", " 1@" e "@1@"	"sinistra@1", "sinistra 1@", "sinistra@1@"	"destra", "destra", "destra"	failed	"sinistra@ ldestra", "sinistra l @destra", "sinistra@ l @destra"	"sinistra@ldestra", "sinistra 1@destra", "sinistra@l@destra"	~	-	~	
21													
	High	Pasquale	1	listOfRightStrings contenente stringhe rispettivamente con " no", "no ", "no [stringa]" e "[stringa]no[stringa]"	"sinistra", "sinistra", "sinistra", "sinistra", "sinistra", "sinistra",	"destra no", "no destra", "nodestra", "desnotra", "destrano ", " nodestra"	passed	"sinistradestra", "sinistradestra", "sinistradestra", "sinistradestra", "sinistradestra", "sinistradestra", "sinistradestra"	"sinistradestra", "sinistradestra", "sinistradestra", "sinistradestra", "sinistradestra", "sinistradestra",				
22													
	Medium	Nicola	1	listOfLeftStrings contenente una stringa con Integer.MAX_VALUE preceduto da un numero	"s1"+String.valueOf (Integer.MAX_VALUE)	"destra"	failed	"s 12147483647destra"	"s12147483647destra"	✓	Se il numero è superiore al massimo intero rappresentabile secondo lo standard IEEE 754, tale numero non viene riconosciuto e quindi il metodo non inserisce lo spazio tra il numero e la restante stringa.	V	
23													
	Medium	Pasquale	1	listOfLeftStrings contenente una stringa con "1 "	"sinistral "	"destra"	passed	"sinistra 1 destra"	"sinistra 1 destra"				Test effettuato per avere il 100% di code coverage
24													
	Medium	Nicola	1	camel case effettuato su un carattere speciale	"sinistra @sinistra sinistra"	"destradestradestra"	passed	"sinistra@sinistraSinistrades tradestradestra"	"sinistra@sinistraSinistradest radestradestra"				Test aggiuntivo scritto dopo aver valutato il codice
25													
	Medium	Pasquale	1	camel case effettuato su un carattere numerico	"sinistra 1sinistra sinistra"	"destradestradestra"	passed	"sinistralSinistraSinistrades tradestradestra"	"sinistralSinistraSinistradest radestradestra"				Test aggiuntivo scritto dopo aver valutato il codice



Test Plan Homework 2

Tester names: Nicola Balzano, Pasquale Di Gennaro

Environment used: Intellij IDEA 2023.2.3

Project: onedrive

PR	IORITY	TESTER NAME	STEP DESCRIPTION			INPUT	STATUS	EXPECTED RESULT	ACTUAL RESULT	BUG FOUND?	BUG DESCRIPTION	SOLVED	COMMENT
	_		_	inputString	firstIndex	lastIndex							
М	ledium	Pasquale	1 firstIndex == lastIndex	"aaa"	2		2 passed	IllegalArgumentException	IllegalArgumentException				
_	ledium	Nicola	1 firstIndex *(-1) == lastIndex	"aaa"	-2		2 passed	IllegalArgumentException	IllegalArgumentException				
_	ledium	Pasquale	1 lastIndex > length(inputString)	"aaa"	1		4 passed	IllegalArgumentException	IllegalArgumentException				
_	ledium	Nicola	1 inputString vuota	""			0 passed	IllegalArgumentException	IllegalArgumentException				
M	Iodium	Pasquale	1 inputetring pull	null			0 passed	IllegalArgumentException	IllegalArgumentException				
171	learum	rasquare	1 inputString null	nuii			passed	TITEGATATGUMENTEXCEPTION	TilegalAlgumentException				
		Nicola	Stringa restituita in ordine crescente con firstIndex pari	charRange('!', '~') && length (inputString)>=0 && length(inputString) <=50	firstIndex >= 0 && firstIndex < length (inputString) && firstIndex % 2 == 0	<pre>lastIndex = firstIndex + length(injectedString)</pre>	passed	injectedString (1000 su 1000 tests)	injectedString (1000 su 1000 tests)				injectedString è formata da charRang ('!', '~') e la sua lunghezza può varia tra 1 e 50
			aggiunti in inputString i caratteri dall' indice 0 all' indice firstIndex della stringa baseString										
	High		ordinamento di injectedString in 2 ordine crescente secondo la conversione ASCII										
			aggiunta injectedString in inputString										
			aggiunti in inputString i caratteri restanti della stringa baseString										
			5 calcolo di lastIndex										
		Pasquale	Stringa restituita in ordine decrescente con firstIndex dispari	charRange('!', '~') && length (inputString)>=0 && length(inputString) <=50	<pre>firstIndex >= 1 && firstIndex < length (inputString) && firstIndex % 2 == 1</pre>	<pre>lastIndex = firstIndex + length(injectedString)</pre>	passed	injectedString (1000 su 1000 tests)	injectedString (1000 su 1000 tests)				injectedString è formata da charRan ('!', '~') e la su lunghezza può vari tra 1 e 50
			aggiunti in inputString i caratteri 1 dall' indice 0 all' indice firstIndex della stringa baseString	-									
	nigh		ordinamento di injectedString in 2 ordine crescente secondo la conversione ASCII										
			aggiunta injectedString in inputString										
			4 aggiunti in inputString i caratteri restanti della stringa baseString										
			5 calcolo di lastIndex										
				,									
		Nicola	Stringa restituita, palindroma e di lunghezza pari con firstIndex negativo	charRange('!', '~') && length (inputString)>=0 && length(inputString) <=50	<pre>firstIndex <= -1 && firstIndex < length (inputString)</pre>	<pre>lastIndex = firstIndex + length(injectedString)</pre>	passed	injectedString (1000 su 1000 tests)	injectedString (1000 su 1000 tests)				injectedString è formata da charRa ('!', '~') e la s lunghezza può var tra 1 e 50
			1 firstIndex = firstindex *(-1)										
	high		aggiunti in inputString i caratteri 2 dall' indice 0 all' indice firstIndex della stringa baseString	-									
			3 modifica di injectedString uguale a se stessa più la stessa invertita										
			<pre>aggiunta injectedString in inputString</pre>										
			5 aggiunti in inputString i caratteri restanti della stringa baseString										
			6 calcolo di lastIndex										

	P	Pasquale	Stringa restituita palindroma e di lunghezza dispari con firstIndex negativo	charRange('!', '~') && length (inputString)>=0 && length(inputString) <=50	firstIndex <= -1 && firstIndex < length (inputString)	<pre>lastIndex = firstIndex + length(injectedString)</pre>	passed	injectedString (1000 su 1000 tests)	injectedString (1000 su 1000 tests)			injectedString è formata da charRange ('!', '~') e la sua lunghezza può variare tra 1 e 50
			1 firstIndex = firstindex *(-1)									
			aggiunti in inputString i caratter: 2 dall' indice 0 all' indice firstIndex della stringa baseString									
1	high		calcolo della stringa invertita nella 3 seconda metà di injectedString senza l'ultimo carattere									
			modifica di injectedString uguale a se 4 stessa più la stringa calcolata al punto 3									
			5 aggiunta injectedString in inputString									
			6 aggiunti in inputString i caratteri restanti della stringa baseString									
			7 calcolo di lastIndex									
0												
Н	High N	Vicola	Restituita stringa data in input	charRange('!', '~') && length (inputString)>=0 && length(inputString) <=50 && uniqueChars && notOrdered	firstIndex <= -1 && firstIndex < length (inputString)	<pre>lastIndex = firstIndex + length(injectedString)</pre>	passed	inputString (1000 su 1000 tests)	inputString (1000 su 1000 tests)			injectedString ha lunghezza 0