

Parameter	Äquivalenzklasse	Repräsentant
char zeichen	gÄk11: [MIN_CHAR,...,MAX_CHAR]	MIN_CHAR
	uÄk11: [NULL,MIN_INT,...MAX_INT]	NULL
char alphabet[]	gÄk21: size of alphabet: [1,..., MAX] membertype: [MIN_CHAR, ..., MAX_CHAR]	3 ['MIN_CHAR','b','MAX_CHAR']
	gÄk22: size of alphabet: [0]	[]
	uÄk21: [NULL], membertype: [NULL]	NULL

Testfall	zeichen	alphabet	Ergebnis
1	MIN_CHAR	['MIN_CHAR','b','MAX_CHAR']	0
2	MIN_CHAR	[]	-1
3	NULL	['MIN_CHAR','b','MAX_CHAR']	NullPointerException
4	MIN_CHAR	NULL	NullPointerException

1b)

Bei den Tests nach DU-Ketten haben wir in Aufgabe 4 nur 2 Tests benötigt wobei wir bei den Äquivalenzklassentest doppelt so viel (4 Tests) benötigten. Auch bzgl. Die Pfadüberdeckung bei DU-Tests ist eine wesentlich höhere als bei dem Äq.Kl. Ansatz. Dies liegt aber auch daran, dass DU-Ketten-Analyse ein Whitebox-Testverfahren im Gegensatz zur blackbox basierten Äquivalenzklassenbildung ist, welche nur an Anhand der Spec testet und nicht den Quellcode betrachtet.