

No.	SubSig	APIPair		RegexWrapper				Relation
		API Invocation caller: str array	CondExpr	R	L	S	Lmin	
1	boolean isEmpty()	r=str.isEmpty()	r==T	$\backslash s\backslash S\{0\}$	-1	NULL	0	≡
2	boolean equals(Object)	r=str.equals(v)	r==T	v	-1	NULL	Len(v)	≡
3	int compareTo(String)	r=str.compareTo(v)	r==0	v	-1	NULL	Len(v)	≡
4	int compareTo(Object)	r=str.compareTo(v)	r==0	v	-1	NULL	Len(v)	≡
5	boolean contentEquals(CharSequence)	r=str.contentEquals(v)	r==T	v	-1	NULL	Len(v)	≡
6	boolean contentEquals(StringBuffer)	r=str.contentEquals(v)	r==T	v	-1	NULL	Len(v)	≡
7	int compareToIgnoreCase(String)	r=str.compareToIgnoreCase(v)	r==0	v	-1	NULL	Len(v)	≠
8	boolean equalsIgnoreCase(String)	r=str.equalsIgnoreCase(v)	r==T	v	-1	NULL	Len(v)	≠
9	boolean contains(CharSequence)	r=str.contains(v)	r==T	$v\backslash s\backslash S^*$	-1	NULL	Len(v)	≠
10	boolean startsWith(String)	r=str.startsWith(v)	r==T	$v\backslash s\backslash S^*$	-1	NULL	Len(v)	≡
11	boolean startsWith(String,int)	r=str.startsWith(v,i ₁)	r==T	$\backslash s\backslash S\{i_1\}v\backslash s\backslash S^*$	-1	NULL	i ₁ +Len(v)	≡
12	boolean endsWith(String)	r=str.endsWith(v)	r==T	$\backslash s\backslash S^*v$	-1	NULL	Len(v)	≡
13	int indexOf(String)	r=str.indexOf(v)	r==i ₁	$\backslash \sim\{i_1\}v\backslash s\backslash S^*$	-1	NULL	i ₁ +Len(v)	≠
		r=str.indexOf(v)	r>=i ₁	$\backslash \sim\{i_1\}v\backslash s\backslash S^*$	-1	NULL	i ₁ +Len(v)	≠
		r=str.indexOf(v)	r<=i ₁	$\backslash \sim\{0,i_1\}v\backslash s\backslash S^*$	-1	NULL	Len(v)	≠
14	int indexOf(int)	r=str.indexOf(v)	r==i ₁	$\backslash \sim\{i_1\}v\backslash s\backslash S^*$	-1	NULL	i ₁ +Len(v)	≡
		r=str.indexOf(v)	r>=i ₁	$\backslash \sim\{i_1\}v\backslash s\backslash S^*$	-1	NULL	i ₁ +Len(v)	≡
		r=str.indexOf(v)	r<=i ₁	$\backslash \sim\{0,i_1\}v\backslash s\backslash S^*$	-1	NULL	Len(v)	≡
				(convert v to string first)				
15	int indexOf(String,int)	r=str.indexOf(v,i ₁)	r==i ₂	$\backslash s\backslash S\{i_1\}\backslash \sim\{i_2-i_1\}v\backslash s\backslash S^*$	-1	NULL	i ₂ +Len(v)	≠
		r=str.indexOf(v,i ₁)	r>=i ₂	$\backslash s\backslash S\{i_1\}\backslash \sim\{i_2-i_1,\}v\backslash s\backslash S^*$	-1	NULL	i ₂ +Len(v)	≠
		r=str.indexOf(v,i ₁)	r<=i ₂	$\backslash s\backslash S\{i_1\}\backslash \sim\{0,i_2-i_1\}v\backslash s\backslash S^*$	-1	NULL	i ₁ +Len(v)	≠
16	int indexOf(int,int)	r=str.indexOf(v,i ₁)	r==i ₂	$\backslash s\backslash S\{i_1\}\backslash \sim\{i_2-i_1\}v\backslash s\backslash S^*$	-1	NULL	i ₂ +Len(v)	≡
		r=str.indexOf(v,i ₁)	r>=i ₂	$\backslash s\backslash S\{i_1\}\backslash \sim\{i_2-i_1,\}v\backslash s\backslash S^*$	-1	NULL	i ₂ +Len(v)	≡
		r=str.indexOf(v,i ₁)	r<=i ₂	$\backslash s\backslash S\{i_1\}\backslash \sim\{0,i_2-i_1\}v\backslash s\backslash S^*$	-1	NULL	i ₁ +Len(v)	≡
				(convert v to string first)				
17	int lastIndexOf(String)	r=str.lastIndexOf(v)	r==i ₁	$\backslash s\backslash S\{i_1\}v\backslash \sim^*$	-1	NULL	i ₁ +Len(v)	≠
		r=str.lastIndexOf(v)	r>=i ₁	$\backslash s\backslash S\{i_1,\}v\backslash \sim^*$	-1	NULL	i ₁ +Len(v)	≠
		r=str.lastIndexOf(v)	r<=i ₁	$\backslash s\backslash S\{0,i_1\}v\backslash \sim^*$	-1	NULL	Len(v)	≠
18	int lastIndexOf(int)	r=str.lastIndexOf(v)	r==i ₁	$\backslash s\backslash S\{i_1\}v\backslash \sim^*$	-1	NULL	i ₁ +Len(v)	≡
		r=str.lastIndexOf(v)	r>=i ₁	$\backslash s\backslash S\{i_1,\}v\backslash \sim^*$	-1	NULL	i ₁ +Len(v)	≡
		r=str.lastIndexOf(v)	r<=i ₁	$\backslash s\backslash S\{0,i_1\}v\backslash \sim^*$	-1	NULL	Len(v)	≡
				(convert v to string first)				
19	int lastIndexOf(String,int)	r=str.lastIndexOf(v,i ₁)	r==i ₂	$\backslash s\backslash S\{i_2\}v\backslash \sim\{i_1+1-i_2-Len(v)\}\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≠
		r=str.lastIndexOf(v,i ₁)	r>=i ₂	$\backslash s\backslash S\{i_2\}v\backslash \sim\{i_1+1-i_2-Len(v)\}\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≠
		r=str.lastIndexOf(v,i ₁)	r<=i ₂	$\backslash s\backslash S\{0,i_2\}v\backslash \sim\{i_1+1-i_2-Len(v)\}\backslash s\backslash S^*$	-1	NULL	i ₁ -i ₂ +1	≠
20	int lastIndexOf(int,int)	r=str.lastIndexOf(v,i ₁)	r==i ₂	$\backslash s\backslash S\{i_2\}v\backslash \sim\{i_1+1-i_2-Len(v)\}\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≡
		r=str.lastIndexOf(v,i ₁)	r>=i ₂	$\backslash s\backslash S\{i_2\}v\backslash \sim\{i_1+1-i_2-Len(v)\}\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≠
		r=str.lastIndexOf(v,i ₁)	r<=i ₂	$\backslash s\backslash S\{0,i_2\}v\backslash \sim\{i_1+1-i_2-Len(v)\}\backslash s\backslash S^*$	-1	NULL	i ₁ -i ₂ +1	≠
21	boolean matches(String)	r=str.matches(v)	r==T	v	-1	NULL	L _{min} (v)	≡
22	boolean regionMatches (int,String,int,int)	r=str.regionMatches(i ₁ ,v ₁ ,i ₂ ,i ₃)	r==T	$\backslash s\backslash S\{i_1\}subStr\backslash s\backslash S^*$ subStr=v ₁ .substring(i ₂ ,i ₂ +i ₃);	-1	NULL	i ₁ +i ₃	≡
23	boolean regionMatches (boolean,int,String,int,int)	r=str.regionMatches(b ₁ ,i ₁ ,v ₁ ,i ₂ ,i ₃)	r==T	$\backslash s\backslash S\{i_1\}subStr\backslash s\backslash S^*$ subStr=v ₁ .substring(i ₂ ,i ₂ +i ₃);	-1	NULL	i ₁ +i ₃	≠
24	char charAt(int)	r=str.charAt(i ₁)	r==i ₂	$\backslash s\backslash S\{i_1\}i_2\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≡
		r=str.charAt(i ₁)	r>=i ₂	$\backslash s\backslash S\{i_1\}[i_2+1]\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≠
		r=str.charAt(i ₁)	r<=i ₂	$\backslash s\backslash S\{i_1\}[i_2-1]\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≠
25	int codePointAt(int)	r=str.codePointAt(i ₁)	r==i ₂	$\backslash s\backslash S\{i_1\}i_2\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≡
		r=str.codePointAt(i ₁)	r>=i ₂	$\backslash s\backslash S\{i_1\}[i_2+1]\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≠
		r=str.codePointAt(i ₁)	r<=i ₂	$\backslash s\backslash S\{i_1\}[i_2-1]\backslash s\backslash S^*$	-1	NULL	i ₁ +1	≠
26	int codePointCount(int,int)	r=str.codePointCount(i ₁ ,i ₂)	r==i ₃	$\backslash s\backslash S\{i_2\}$	-1	NULL	i ₂	≡
		r=str.codePointCount(i ₁ ,i ₂)	r>=i ₃	$\backslash s\backslash S\{i_2\}$	-1	NULL	i ₂	≠
		r=str.codePointCount(i ₁ ,i ₂)	r<=i ₃	$\backslash s\backslash S\{i_2\}$	-1	NULL	i ₂	≠
27	int codePointBefore(int)	r=str.codePointBefore(i ₁)	r==i ₂	$\backslash s\backslash S\{i_1-1\}i_2\backslash s\backslash S^*$	-1	NULL	i ₁	≡
		r=str.codePointBefore(i ₁)	r>=i ₂	$\backslash s\backslash S\{i_1-1\}[i_2+1]\backslash s\backslash S^*$	-1	NULL	i ₁	≠
		r=str.codePointBefore(i ₁)	r<=i ₂	$\backslash s\backslash S\{i_1-1\}[i_2-1]\backslash s\backslash S^*$	-1	NULL	i ₁	≠
				(convert i ₂ -1,i ₂ +1,i ₂ to string first)				
28	int offsetByCodePoints(int,int)	r=str.offsetByCodePoints(i ₁ ,i ₂)	r==i ₃ (>=,<=,>,<)	$\backslash s\backslash S\{i_1+i_2,\}$	-1	NULL	i ₁ +i ₂	≡
29	int length()	r=str.length()	r==i ₁	$\backslash s\backslash S\{i_1\}$	-1	NULL	i ₁	≡
		r=str.length()	r>=i ₁	$\backslash s\backslash S\{i_1,\}$	-1	NULL	i ₁	≡
		r=str.length()	r<=i ₁	$\backslash s\backslash S\{0,i_1\}$	-1	NULL	0	≡
30	char[] toCharArray()	r=str.toCharArray()	r[i]==v	$\backslash s\backslash S\{i\}v\backslash s\backslash S^*$	-1	NULL	i+1	≡
		r=str.toCharArray()	r[i]>v	$\backslash s\backslash S\{i\}[v+1]\backslash s\backslash S^*$	-1	NULL	i+1	≠
		r=str.toCharArray()	r[i]<v	$\backslash s\backslash S\{i\}[v-1]\backslash s\backslash S^*$	-1	NULL	i+1	≠
				(convert v+1,v-1 to string first)				
31	byte[] getBytes(String)	r=str.getBytes(v ₁)	r[i]==v	$\backslash s\backslash S\{i\}v\backslash s\backslash S^*$	-1	NULL	i+1	≡
		r=str.getBytes(v ₁)	r[i]>v	$\backslash s\backslash S\{i\}[v+1]\backslash s\backslash S^*$	-1	NULL	i+1	≠
		r=str.getBytes(v ₁)	r[i]<v	$\backslash s\backslash S\{i\}[v-1]\backslash s\backslash S^*$	-1		i+1	≠
				(convert v+1,v-1 to string first)		NULL		
32	byte[] getBytes(Charset)	r=str.getBytes(v ₁)	r[i]==v	$\backslash s\backslash S\{i\}v\backslash s\backslash S^*$	-1	NULL	i+1	≡
		r=str.getBytes(v ₁)	r[i]>v	$\backslash s\backslash S\{i\}[v+1]\backslash s\backslash S^*$	-1	NULL	i+1	≠
		r=str.getBytes(v ₁)	r[i]<v	$\backslash s\backslash S\{i\}[v-1]\backslash s\backslash S^*$	-1	NULL	i+1	≠
				(convert v+1,v-1 to string first)				
33	byte[] getBytes()	r=str.getBytes()	r[i]==v	$\backslash s\backslash S\{i\}v\backslash s\backslash S^*$	-1	NULL	i+1	≡
		r=str.getBytes()	r[i]>v	$\backslash s\backslash S\{i\}[v+1]\backslash s\backslash S^*$	-1	NULL	i+1	≠
		r=str.getBytes()	r[i]<v	$\backslash s\backslash S\{i\}[v-1]\backslash s\backslash S^*$	-1	NULL	i+1	≠
				(convert v+1,v-1 to string first)				
34	String substring(int)	r=str.substring(i ₁)	NULL	$\backslash s\backslash S\{i_1\}$	-1	NULL	i ₁	≡
35	String substring(int,int)	r=str.substring(i ₁ ,i ₂)	NULL	$\backslash s\backslash S\{i_1\}$	i ₂ -i ₁	NULL	i ₁	≠
36	CharSequence subSequence(int,int)	r=str.subSequence(i ₁ ,i ₂)	NULL	$\backslash s\backslash S\{i_1\}$	i ₂ -i ₁	NULL	i ₁	≠
37	String[] split(String)	r=str.split(v)	r[i]	$\backslash v\{i\}$	-1	$\backslash \sim$	i	≠
38	String[] split(String,int)	r=str.split(v, i ₁)	r[i]	$\backslash v\{i\}$	-1	$\backslash \sim$	i	≠
39	String replace(char,char)	r=str.replace(v ₁ ,v ₂)	NULL	NULL	-1	$\backslash \sim v_1$	0	≠
40	String replace(CharSequence,CharSequence)	r=str.replace(v ₁ ,v ₂)	NULL	NULL	-1	$\backslash \sim v_1$	0	≠
41	String replaceAll(regex,String)	r=str.replaceAll(v ₁ ,v ₂)	NULL	NULL	-1	$\backslash \sim v_1$	0	≠
42	String replaceFirst(String, String)	r=str.replaceFirst(v ₁ ,v ₂)	NULL	NULL	-1	$\backslash \sim v_1$	0	≠
43	String toLowerCase(Locale)	r=str.toLowerCase(local)	NULL	NULL	-1	NULL	0	≠
44	String toLowerCase()	r=str.toLowerCase()	NULL	NULL	-1	NULL	0	≠
45	String toUpperCase(Locale)	r=str.toUpperCase(local)	NULL	NULL	-1	NULL	0	≠
46	String toUpperCase()	r=str.toUpperCase()	NULL	NULL	-1	NULL	0	≠
47	String toString()	r=str.toString()	NULL	NULL	-1	NULL	0	≠
48	String trim()	r=str.trim()	NULL	NULL	-1	$\backslash \sim$	0	≠
new add:	Array: int length	r=arr.length	r ==i	$\backslash s\backslash S\{i\}$	-1	NULL	i	≡
		r=arr.length	r >=i	$\backslash s\backslash S\{i,\}$	-1	NULL	i	≡
		r=arr.length	r <=i	$\backslash s\backslash S\{0,i\}$	-1	NULL	0	≡