

Section 1: Regular Expressions

Regular Expression Reference Table

Symbol	Meaning	Example(s)
a	Literal character	a , any symbol in your alphabet Σ
ab	Concatenation of regex's (or literal characters) a and b	ab , $(abc)(def)$
ε	Empty string	ε
$a b$	a or b	a, b
a^*	0 or more a 's	$\varepsilon, a, aaaa, aaaaa$
a^+	1 or more a 's	$a, aaa, aaaaa$
$a?$	0 or 1 a 's	ε, a
$[a-z]$	1 character in range a - z : ($a b \dots z$)	$a, b, c, d, e, f, \dots, z$
$[skj]$	1 of characters in bracket: ($s k j$)	s, k, j

1) Describe the meaning of each of the following regular expressions in English and give two different strings it can produce:

i) $(1 | 0)^* 0$

Non-empty binary strings ending with 0

ii) $([A-Z][a-z]^* | [0-9]^+)$

Nonempty sequence of letters with first letter upper case or sequence of base 10 digits

iii) $(\varepsilon | 4?0^+1^* X 3^+)$

Two strings that can be produced: 401 X 333 or empty string

2) Write a regular expression for each of the following specifications:

i) All strings consisting of 0's and 1's (binary digits) with an even number of 0s

$1^* (0 1^* 0 1^*)^*$

ii) camelCased variable name in Java, where the alphabet is upper and lower-cased letters without any numbers or underscores

$[a-z]^+([A-Z][a-z]^*)^*$

iii) Non-empty binary strings where each 1 directly follows a 0 (challenge 1: only use symbols in table up until $*$; challenge 2: also avoid $|$.)

Challenge 1: $(0 | 01)^* (0 | 01)$

Challenge 2 (no or): $0 (0^* (0 1))^* 0^*$

Normal: $(0+1?)+$

Addendum: as noted in Thursday's sections, "Challenge 2" answer above is incorrect, since it omits "01". Removing the initial zero fixes that but allows ε . I don't see a solution that fixes both issues.