

1. Genere las siguientes clases de caracteres.

- a) Todas las letras que no sean vocales.

`[b-df-hj-np-tv-z]`

- b) Todas las letras que sean vocales

`[aeiouAEIUO]`

- c) Los dígitos hexadecimales

`[0-9a-fA-F]`

- d) Los dígitos octales

`[0-7]`

- e) Los operadores aritmeticos

`[-+*/%]`

2. Genere las expresiones regulares para los siguientes lenguajes.

- a) Los números hexadecimales en C.

`0[xX][0-9a-fA-F]+`

- b) Los números octales en C.

`0[0-7]*`

- c) Los comentarios del lenguaje C que no pueden contener `*/` intermedio.

`“*/”([^[*/])([*/])“*/”`

- d) Los números binarios múltiplos de cuatro.

`[01]*[1]+[01]*00`

- e) Los números complejos.

entero: $[0-9]^+$
 unidad_imaginaria: $[ij]$
 signo: $[+-]$
 decimal: entero . entero?
 | entero? . entero
 exponente: $[eE]$ signo? entero
 real: decimal exponente?
 complejo: signo? real
 | signo? real? signo decimal unidad_imaginaria
 | signo? real? signo unidad_imaginaria real
 | decimal unidad_imaginaria
 | unidad_imaginaria real

f) Sobre el alfabeto $\{a, b, c, d\}$ todas las cadenas que no contienen la subcadena adbc.

$(b|c|d)^*(a|b|c)^*(a|d|c)^*(a|b|d)^*$

g) Todas las cadenas sin dígitos repetidos.

No

h) El lenguaje de las direcciones IP válidas.

segmento: $(25[0-5]|2[0-4][0-9]|1[0-9][0-9]|[1-9]?[0-9])$
 ip: segmento.segmento.segmento.segmento

i) Todas las cadenas de números pares.

$[0-9]^*[02468]$

j) Todas las cadenas de números impares.

$[0-9]^*[13579]$

k) Todas las cadenas de números que no tengan ceros al principio excepto cero.

$0|[1-9]+[0-9]^*$

l) Las palabras reservadas INSERT, SELECT, FROM, escritas en cualquier combinación de minúsculas y mayúsculas.

$[Ii][Nn][Ss][Ee][Rr][Tt][Ss][Ee][Ll][Ee][Cc][Tt][Ff][Rr][Oo][Mm]$

3. De las siguientes expresiones regulares diga cuáles si generam correctamente los comentarios en lenguaje C.

- a) `"/*" (~\epsilon) *"/`
- b) `"/*" [^* /] *"/`
- c) `"/*" (("/") * [^* /] | "/") * ("") + "/"`