Instrucciones:

Tus respuestas a esta Actividad deben de ponerse en un documento de Word, para subirlo a Canvas.

Como ejemplo de lo que espero, usa como referencia los archivos PDF de Clase01 y Clase02 en Canvas. Como ves en esos dos PDF, no hay nada escrito a mano.

Es decir, no incluyas tus respuestas contestadas a mano, ni tampoco incluyas imágenes de tus respuestas a mano dentro del documento Word. En caso de no cumplir con este requerimiento, tu Actividad no se tomará en cuenta y tendrás cero puntos en esta Actividad.

No está por demás recordarte que si copias o dejas que te copien, tendrás cero puntos y muy posiblemente un reporte de deshonestidad académica.

Problema 1.

**Considera X = {aaa,bbb} y Y = {λ, a, ba}.**

1. **Lista los strings en el conjunto XY.**

* xy= {aaa,bbb,aaaa,bbba,aaaba,bbbba}

1. **¿Cuántos strings de largo 6 se encuentran en X\*?**

{aaaaaa,bbbbb,aaabbb,bbbaaa}

* 4

1. **Lista los strings en el conjunto Y\* de largo tres o menos.**

* **{** λ **,a,aa,ba,aba,baa,aaa}**

**d)      Lista los strings en el conjunto X\*Y\* de largo cuatro o menos.**

**X {aaa,bbb}**

* **Y {** λ **,a,aa,ba,aba,baa,aaa,aaaa,abaa,baba,baaa}**
* **conjunto X\*Y\***
* **aaaa, bbba,**

Problema 2.

Da un definición recursiva del conjunto de strings a partir de Σ={a,b} en el cual cada string comienza con una b y tiene una longitud múltiplo de tres. Ejemplo de definición recursiva la puedes ver en el Ejemplo 3 de la Clase07.

1. Base bbb, bab ∊L.
2. Paso recursivo: Si u ∊L, entonces ubbb,ubab,ubba,uaaa ∊L.

Problema 3.

Haz una expresión regular con la nomenclatura de Racket, como se vio en los ejemplos en la segunda mitad de Clase 08.

Te puede resultar útil utilizar una funcionalidad adicional que Racket agrega a las expresiones regulares para abreviar las expresiones. [0-9] es una abreviación de (0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9).

La expresión regular debe de encontrar strings con las siguientes características.

Debe de buscar por enteros. Los enteros deben tener al menos un dígito. También debe de reconocer el signo + o – que lo inmediatamente lo precede, en caso de que el signo esté presente, indicando explícitamente que es un numero positivo o negativo, respectivamente.

Ejemplos:

45

-786

+4

Problema 4.

Haz una expresión regular con la nomenclatura de Racket, como se vio en los ejemplos en la segunda mitad de Clase 08.

Te puede resultar útil utilizar una funcionalidad adicional que Racket agrega a las expresiones regulares para abreviar las expresiones. [0-9] es una abreviación de (0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9).

La expresión regular debe de encontrar strings con las siguientes características.

Debe de buscar por número reales. Estos reales NO usan la notación científica. Al lado izquierdo del punto decimal puede tener cero o más dígitos. Al lado derecho del punto decimal puede tener uno o más dígitos. También debe de reconocer el signo + o – que lo inmediatamente lo precede, en caso de que el signo esté presente, indicando explícitamente que es un numero positivo o negativo, respectivamente.

Ejemplos:

.45

-.86

+4.0

3.1416

32.987

Problema 5.

Haz una expresión regular con la nomenclatura de Racket, como se vio en los ejemplos en la segunda mitad de Clase 08.

Te puede resultar útil utilizar una funcionalidad adicional que Racket agrega a las expresiones regulares para abreviar las expresiones. [0-9] es una abreviación de (0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9).

La expresión regular debe de encontrar strings con las siguientes características.

Debe de buscar por número reales. Estos reales SI usan la notación científica. Al lado izquierdo del punto decimal puede tener cero o más dígitos. Al lado derecho del punto decimal puede tener uno o más dígitos. También debe de reconocer el signo + o – que lo inmediatamente lo precede, en caso de que el signo esté presente, indicando explícitamente que es un numero positivo o negativo, respectivamente.

Ejemplos:

.45

-.86

+4.0

3.1416

32.987

.45e2

-.86e-3

+4.0E+5

3.1416E-2

32.987E3

**Problema 6.**

**La siguiente tabla tiene identidades de expresiones regulares.**

|  |  |
| --- | --- |
| 1 | Øu = uØ = Ø |
| 2 | λu = uλ = u |
| 3 | Ø\* = λ |
| 4 | λ\*= λ |
| 5 | u U v = v U u |
| 6 | (u U v) U w = u U (v U w) |
| 7 | (uv)w = u(vw) |
| 8 | u U Ø = u |
| 9 | u U u = u |
| 10 | u\* = (u\*)\* |
| 11 | u+ = uu\* |
| 12 | u(v U w) = uv U uw |
| 13 | (v U w)u = vu U wu |
| 14 | (uv)\*u = u(vu)\* |
| 15 | (u U v)\* = (u\* U v)\*  = u\*(u U v)\*  = (u U vu\*)\*  = (u\*v\*)\*  = u\*(vu\*)\*  = (u\*v)\*u\* |

**Usa las identidades en esta tabla para comprobar las siguientes igualdades:**

1. **(ba)+(a\*b\* U a\*) = (ba)\*ba+(b\* U λ)**

a\*(b\*U λ)

(ba) + (a+ · λ) (b\*U λ)

(ba)\* · b · a + (b\* U λ)

* (ba)\* · b· a + (b\*U λ)

1. **b+(a\*b\* U λ)b = b(b\*a\* U λ)b+**

b(b\*a\* U λ)b+

* b (b\*a\* U λ) b+