**Teoría Computacional**

**Práctica 2. Operaciones en lenguajes**

**Objetivo:** Hacer la demostración de las operaciones en lenguajes vistas en clase, mediante un programa escrito en Java, C, o algún otro lenguaje de programación.

**Desarrollo:** El programa debe hacer lo siguiente:

a) Leer el alfabeto **Σ** que servirá como base para resolver esta práctica. El alfabeto debe tener al menos tres símbolos. Los símbolos deben poder ser ingresados de dos maneras posibles:

* En forma individual (de uno en uno).
* En forma de enumeración (a-z ó d-m ó 0-9 ó 3-8, etc.) ingresando el primer símbolo del alfabeto y a continuación el último símbolo del mismo.

b) Con base en el alfabeto **Σ** generar los lenguajes **L1** y **L2** de forma aleatoria. El número de elementos o palabras (p) a ser generados así como su longitud (l), serán valores de entrada que ingresará el usuario. Desplegar en pantalla los lenguajes generados **L1** y **L2**.

c) Generar el lenguaje **LU** resultado de la **unión** de **L1** y **L2**. Desplegar en pantalla **LU**.

d) Generar el lenguaje **LC** resultado de la **concatenación** de **L1** y **L2**. Desplegar en pantalla **LC**.

e) Generar el lenguaje **LD1**, resultado de la **diferencia** de **L1** – **L2**; y **LD2**, resultado de la **diferencia** de **L2** – **L1**. Mostrar ambos resultados en pantalla.

f) Generar el lenguaje **LP** resultado de la **potencia** del lenguaje **L1** o **L2** (habilitar una opción para seleccionar el lenguaje). La potencia es un valor entre -5 y +5, este valor se debe proporcionar como dato de entrada. Desplegar en pantalla **LP**

g) Realiza una de las siguientes aplicaciones:

* Generar una CURP. Hacerlo de forma automática y con datos aleatorios; la CURP generada debe tener información válida (RFC válido, el campo de sexo debe ser H o M, la entidad federativa debe corresponder con la lista oficial, etc.).
* Generar una placa de circulación de automóvil (con base en el ejercicio de modelación de lenguajes visto en clase)
* Tema libre (debe reunir un grado de complejidad semejante a los dos ejemplos anteriores)

**Notas para el inciso g:**

1. La aplicación debe generar un dato válido, de forma aleatoria, y conforme a los requerimientos de cada caso.

2. La aplicación debe estar basada en la modelación de **alfabetos**, **lenguajes**, **palabras,** esto es, conceptos vistos en clase que sean aplicables a esta práctica. Por lo tanto, esta modelación debe estar contemplada en el desarrollo.

**Evaluación**:

* Incisos a y b: un punto cada uno.
* Incisos c, d, e, f: un punto y medio cada uno.
* Inciso g: dos puntos

**Presentación de la práctica:**

* Presentar el programa en ejecución, mostrando cada uno de los incisos (o los que tengan resueltos).
* Sustentar un breve examen oral acerca del código y de los conceptos de Teoría Computacional empleados en esta práctica.
* No es necesario entregar reporte escrito.
* Prácticas copiadas serán canceladas.
* Presentar práctica en laboratorio, en salón de clase, o en cubículo de profesor (previa cita).
* Individual o en equipo (dos personas máximo).
* No enviar la práctica por correo electrónico. Las prácticas enviadas por este medio serán canceladas.

**Fecha de entrega:**

Fecha de caducidad de esta práctica: viernes 15 de septiembre de 2017. Trabajos demorados en el lapso de una semana: 50% menos en la calificación; no se aceptarán prácticas después de este lapso.