

# Teoría Computacional

## Práctica 1. Alfabetos, lenguajes y expresiones regulares

### Objetivo.

Hacer la implementación de alfabetos y lenguajes, conceptos vistos en clase, mediante un lenguaje de programación de alto nivel: Java, C++, C, o algún otro.

### Desarrollo.

1. Leer el alfabeto  $\Sigma_1$  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:
  - De forma individual (de uno en uno, o de dos en dos, dependiendo de la longitud de cada símbolo), por ejemplo: a, b, c, d, e, f; o bien: a1, b4, c5.
  - Por rango, ingresando el primer símbolo del alfabeto a definir y a continuación el último (por ejemplo: a-z, m-x, F-Q, 0-9, 5-9, etc.).
2. Leer dos cadenas:  $w_1$  y  $w_2$  ambas elementos del alfabeto  $\Sigma_1$ . Las cadenas deben ser validadas por el programa: en caso de error en el ingreso de las cadenas, se debe hacer la indicación al usuario para que vuelva a ingresar la cadena de forma correcta. Una cadena es inválida si contiene algún símbolo que no pertenezca al alfabeto.
3. Indicar si  $w_1$  es un prefijo o sufijo (propio o no propio), o subcadena, o subsecuencia, o cualquier combinación anterior, de  $w_2$ .
4. Con base en el alfabeto  $\Sigma_1$  generar los lenguajes  $L_1$  y  $L_2$  de forma aleatoria. El número de elementos o palabras (np) a ser generados así

como su longitud ( $|$ ), serán valores de entrada que ingresará el usuario. Desplegar en pantalla los lenguajes generados  $L_1$  y  $L_2$ .

5. Generar el lenguaje  $LU$  resultado de la **unión** de  $L_1$  y  $L_2$ . Desplegar en pantalla  $LU$ .
6. Generar el lenguaje  $LC$  resultado de la **concatenación** de  $L_1$  y  $L_2$ . Desplegar en pantalla  $LC$ .
7. Mediante la utilización de expresiones regulares y/o definiciones regulares, implementar **uno** de los siguientes incisos:
  - A. Todas las cadenas de letras en minúsculas (a-z) que contengan las cinco vocales en orden.; las vocales pueden estar repetidas (siempre que mantengan el orden), las secuencias de las vocales también pueden repetirse. Ejemplos de palabras aceptadas:  
*rtaeioujutf, artehjiyibgfohgfdujhfd, aejhjihgghgougha, hknalleioomuwraamqekiodsu.*
  - B. Todas las cadenas de dígitos que tengan por lo menos un dígito repetido. Los dígitos no tienen que estar en orden. Ejemplos de cadenas aceptadas: *12345672, 1286182, 1821.*
  - C. Los identificadores para funciones y variables en lenguaje C; o los identificadores para clases, métodos y variables en Java.

**Nota:** En el inciso 7, el programa debe solicitar al usuario que ingrese la palabra a analizar. Como respuesta, el programa desplegará en pantalla la leyenda “palabra correcta” si la palabra ingresada cumple con la expresión regular; de otro modo, desplegará “palabra incorrecta” lo cual indica que la palabra ingresada no cumple con la expresión regular.

## Evaluación.

Concepto	Valor
Generación del alfabeto	2
Generación de dos palabras válidas del alfabeto	1
Prefijos, sufijos, subcadenas, subsecuencias	1
Generación de dos lenguajes	2
Unión de lenguajes	1
Concatenación de lenguajes	1
Expresiones regulares	2

## Presentación de la práctica.

- La práctica se puede desarrollar en equipo con un máximo de dos personas, o de manera individual.
- La presentación se hará mediante una conferencia virtual en donde el equipo, o alumno(a) de forma individual, presentará el programa en ejecución. Se asignarán los horarios conforme a la disponibilidad de horario de la clase y también fuera de este horario.
- Se hará un breve examen oral (dos o tres preguntas) acerca del código y también de conceptos de Teoría Computacional en el programa. Para el caso de los equipos, las preguntas se harán de forma aleatoria a cada integrante. Si el alumno(a) reprueba el examen oral, también reprueba la práctica aunque el código cumpla con los requerimientos de la especificación.
- Se dará la calificación de la práctica en el momento en que concluya la presentación de la misma; entonces deberán subir la práctica a la plataforma *Teams* para que quede como evidencia.
- No se aceptarán prácticas que no hallan sido revisadas, aunque se suban a la plataforma en el tiempo especificado.
- Las prácticas copiadas serán canceladas.

## **Fechas de Entrega.**

La práctica 1 se revisará en la semana del 20 al 24 de septiembre. Los horarios de entrega se asignarán la semana previa, durante las horas de clase. No habrá prórroga en la entrega, no se reasignará horario si el equipo o alumno no se presenta en la hora programada; tampoco se aceptarán prácticas enviadas por correo electrónico a mi cuenta institucional (IPN).