

# ICI-142 Fundamentos de Programación

## Tarea #3

Makarena Donoso <makarena.donos@gmail.com>

Wenceslao Palma <wenceslao.palma@ucv.cl>

Claude Shannon, científico fallecido en febrero de 2001, creó las bases matemáticas de la teoría de la información. En su trabajo fundamental, *A Mathematical Theory of Communication* propuso una medida llamada *Entropía*.

En esta tarea usaremos el concepto de *entropía* para analizar textos a nivel de su variedad de palabras.

Se define la *entropía* de un texto  $T$ , con  $\lambda$  palabras de las cuales  $n$  de ellas son distintas, mediante :

$$E_T(p_1, \dots, p_n) = \frac{1}{\lambda} \sum_{i=1}^n p_i [\log_{10}(\lambda) - \log_{10}(p_i)]$$

donde

$p_i$ , es la frecuencia de la  $i$ -ésima palabra en el texto  $T$ , es decir,  $p_i$  es el número de veces que la  $i$ -ésima palabra aparece en el texto.

Si consideramos que un texto de largo  $\lambda$  tiene más riqueza cuando la cantidad de palabras distintas es grande y que entre textos con el mismo  $\lambda$  y el mismo  $n$  tiene más riqueza aquél que tiene menos variación en su frecuencia, es posible concluir que la *entropía* es una medida muy útil para comparar la riqueza de 2 o más textos. Para comparar textos con diferente cantidad de palabras, introduciremos una entropía relativa  $E_{rel}$  definida como :

$$E_{rel} = \frac{E_T}{E_{max}} * 100$$

donde :

$$E_{max} = \frac{1}{\lambda} \sum_{i=1}^n 1 [\log_{10}(\lambda) - \log_{10}(1)] = \log_{10}(\lambda)$$

Escriba un programa en C que calcule para un texto  $T$ :  $\lambda, E_T$  y  $E_{rel}$ . El programa no debe distinguir entre mayúsculas y minúsculas. Además una palabra es una secuencia consecutiva de caracteres diferentes a `, . ; ! ? " ( )` así como también espacios, tabs y newline. Palabras compuestas de un caracter tambien son consideradas.

**Datos de entrada** : el programa deberá permitir el ingreso de dos textos  $T$ , con  $1 \leq \lambda$ . El largo máximo para una palabra es 20 caracteres y un texto no contiene mas de 7 palabras por línea, además la cantidad de líneas para un texto no es mayor a 10. Es posible que el texto contenga líneas en blanco. El ingreso debe ser línea por línea con ajuste automático.

Ambos textos deben ser almacenados en arreglos bidimensionales.

**Salida:** Para cada texto se debe mostrar  $\lambda$ ,  $E_T$  redondeado a un dígito decimal y  $E_{rel}$  en porcentaje redondeado a un entero. Ejemplo:

$T_1$ :  $\lambda_1$ ,  $E_{T_1}$ ,  $E_{rel_1}$

$T_2$ :  $\lambda_2$ ,  $E_{T_2}$ ,  $E_{rel_2}$

**Restricciones:**

- La tarea debe ser codificada en Lenguaje C. No utilice funciones que no pertenecen al ANSI C
- solo se consideran programas que cumplan con la utilización de funciones, arreglos bidimensionales y las especificaciones para los datos de entrada y salida.
- Solo se recibirán tareas fuera de plazo dentro de las 24 horas siguientes a la fecha de entrega. Nota máxima es un 5.0.

Fecha de entrega : Miércoles 20 de Octubre, código fuente tarea3.c indicando nombre y rol, enviar por email hasta las 24h00.

email grupo Prof. Wenceslao Palma <ici142-01@inf.ucv.cl>

email grupo Prof. Makarena Donoso <ici142-02@inf.ucv.cl>