

## 2.1. Secuencias de datos elementales

### Ejercicio 2.1.1

---

Dada una secuencia TEXTO construir un algoritmo que cuente e informe por oración cuántas palabras terminan con la sílaba "LA".

### Ejercicio 2.1.2

---

Dada una secuencia de caracteres que contiene un texto compuesto por oraciones, crear una secuencia de salida que contenga sólo las palabras que comienzan con la sílaba "MA". Informar para cada oración cuántas palabras cumplen con la condición.

### Ejercicio 2.1.3

---

Dada una secuencia de caracteres que contiene un texto compuesto por oraciones; crear una secuencia de salida que contenga solo las palabras con 4 letras, informar para cada oración cuantas palabras cumplen con la condición. El fin de cada oración esta dado por un ":".

### Ejercicio 2.1.4

---

Dada una secuencia de texto de entrada, desarrollar un algoritmo que produzca una secuencia oración de salida; la que contendrá palabras numéricas formadas con los dígitos impares que contenga la palabra correspondiente de entrada . Ej. Si la entrada viniera "A125EB%" en la salida debería ir "15". Al final del proceso informar cuál es el promedio de palabras por oración encontradas en la secuencia de entrada.

### Ejercicio 2.1.5

Escribir un algoritmo que: dado un fichero de texto que contiene letras y números, permita distinguir palabras de enteros de hasta 5 cifras y desglosarlos en dos secuencias que contengan respectivamente las palabras pares y las impares.

### Ejercicio 2.1.6

---

Escribir un algoritmo que permita desglosar una secuencia que contiene monotonías crecientes (series de elementos consecutivos ordenados crecientemente; cada serie está constituida por un elemento como mínimo), en dos secuencias de salida, de tal modo que tales monotonías sean copiadas alternativamente en una de las secuencias resultado o en otra.

### Ejercicio 2.1.7

---

Dada una secuencia texto de entrada, desarrollar un algoritmo que produzca una secuencia oración de salida; la que contendrá palabras numéricas formadas con los dígitos pares que contenga las palabras correspondientes de entrada. Ej.: entrada (q12w34e58) → salida (248). Al final del proceso informar cual es el promedio de palabras por oración encontradas en la secuencia de entrada.

### Ejercicio 2.1.8

---

Escribir un algoritmo que a partir de una secuencia de palabras dada, genere dos secuencias de salida. Una de ellas contendrá solo las palabras de la secuencia original que comienzan con vocal y la otra será una secuencia numérica en la que se almacenarán las longitudes de las palabras que cumplieron tal condición. Por final de proceso se deberá informar la cantidad de palabras de ambas secuencias de palabras y la longitud de la palabra más larga que cumplió la condición enunciada.

### Ejercicio 2.1.9

---

Escribir un algoritmo que a partir de una secuencia de palabras dada, genere una secuencia de igual tipo contenido solo las palabras de la secuencia original que NO comienzan con vocal . Por final de proceso se deberá informar la cantidad de palabras de ambas secuencias de palabras y el porcentaje de palabras que no cumplieron con la condición enunciada.

## Ejercicio 2.1.10

---

Escribir un algoritmo que permita efectuar la operación de diferencia entre dos secuencias (ordenadas crecientes), esto es, que genere el conjunto de los elementos no comunes a las dos secuencias.

## Ejercicio 2.1.11

---

Escribir un algoritmo que dado dos ficheros de texto (caracter, palabra, oraciones) denominadas TEX1 y TEX2 genere otro fichero oración, mediante la composición de palabras en posiciones pares dentro de las oraciones del TEX1 y palabras en posiciones impares dentro del TEX2. Al final del proceso emitir los siguientes contadores:

1. Cantidad de palabras del archivo de salida.
2. Promedio de palabras por oración de TEX1 Y TEX2.

## Ejercicio 2.1.12

---

Dada una secuencia de caracteres con datos de personas con el siguiente formato:

```
Nombre | Domicilio | DNI (8 caracteres) Región (1-2-3) Nombre | Domicilio | DNI (8  
caracteres) Región (1-2-3).....\*(marca de fin)
```

DNI, Región y Nombre de la siguiente persona no están separados por ningún carácter, Región tiene un solo carácter.

Generar una secuencia de caracteres de salida que contenga todos los DNI que comiencen con dígito par y pertenezcan a la Región 1.

Al final informar cantidad de personas de cada región y porcentaje de las mismas sobre el total de personas

## Ejercicio 2.1.13

---

Se tiene una secuencia de caracteres que finaliza con el carácter "\*", el problema es que este carácter puede aparecer dentro de la secuencia, para poder distinguirlo con el fin de secuencia se utiliza otro símbolo especial "\\" (carácter de escape). Se pide realizar un algoritmo que permita

escribir en una secuencia de salida todos los caracteres excepto los de escape, y contar la cantidad de escapes que existen.

### Ejercicio 2.1.14

---

Dada una secuencia de caracteres de entrada, desarrollar un algoritmo que produzca una secuencia de salida de caracteres; la que contendrá sólo los caracteres “numéricos” correspondientes a dígitos impares encontrados en la secuencia de entrada.

Al final del proceso informar total de caracteres “numéricos” (dígitos) y porcentaje de pares e impares sobre el total de caracteres numéricos

### Ejercicio 2.1.15

---

Dada una secuencia de caracteres de entrada, desarrollar un algoritmo que produzca dos secuencias de salida de caracteres. La primera secuencia de salida contendrá los caracteres numéricos pares y la segunda secuencia de salida contendrá los caracteres numéricos impares. Al final del proceso calcular e informar el porcentaje de caracteres numéricos pares e impares sobre el total de caracteres numéricos.

### Ejercicio 2.1.16

---

Dada una secuencia de caracteres de entrada, desarrollar un algoritmo que produzca dos secuencias de salida de caracteres; en donde la primera contendrá todos los caracteres numéricos y la segunda los demás. Al final del proceso calcular e informar el porcentaje de caracteres numéricos sobre el total.

### Ejercicio 2.1.17

---

Escribir un algoritmo que permita desglosar una secuencia de enteros que contiene monotonías crecientes (series de elementos consecutivos ordenados crecientemente; cada serie está constituida por un elemento como mínimo), en dos secuencias de salida, de tal modo que tales monotonías sean copiadas alternativamente en una de las secuencias resultado o en otra.

Ejemplo secuencia de entrada

2, 5, 8, 11, 14, 17, 15, 10, 11, 20, 44, 12, 14, 16, ...

#### Secuencia de salida 1

2, 5, 8, 11, 14, 17, 10, 11, 20, 44, ...

#### Secuencia de salida 2

15, 12, 14, 16, ...

Obtener la suma de cada monotonía e informar aquella de mayor valor.

### Ejercicio 2.1.18

---

Se dispone de 2 secuencias de caracteres, una contiene caracteres, incluidos espacios y otra contiene caracteres “numéricos”. Se pide crear una nueva secuencia de salida de caracteres donde se copien los caracteres de ambas secuencias en forma alternada, o sea un carácter, un dígito, un carácter, un dígito, etc.

Ejemplo: Sec1: Para habilitar la función de Salas para grupos pequeños para su uso\*

Sec2: 13481460918460846161334868461086\*

Salida: P1a3r4a8 1h4...

1. Analice y responda: Que pasaría en el caso de que una secuencia de entrada termine antes que la otra? se podría cumplir con la consigna?
2. Como modificaría el algoritmo en el caso de que solicite contar la cantidad de caracteres de cada secuencia y mostrar por pantalla al final del proceso?