



## Trabajo Práctico Nº 9

### División de Problemas

#### Ejercicio 1 (*Parcial 2015*):

El archivo **alumnos.txt** contiene los nombres y notas de un grupo de alumnos. Un nombre siempre aparece seguido de un espacio, el apellido seguido de un espacio y a continuación la nota correspondiente.

Por ejemplo:

```
Alvaro Perez 4 <eoln>
Juan Gomez 6 <eoln>
Carlos Suarez 9 <eoln>
Diego Ramos 10 <eoln>
Edgardo Lopez 2 <eoln>
<eof>
```

Escribir un programa dividiendo el problema en subproblemas para generar un archivo de texto **conceptos.txt** que contenga: la inicial del nombre de cada alumno, y luego el apellido seguido de su concepto ("APROBADO" si la nota es mayor a 5, "DESAPROBADO" en caso contrario).

Ejemplo del archivo generado:

```
A Perez Desaprobado <eoln>
J Gomez Aprobado <eoln>
C Suarez Aprobado <eoln>
D Ramos Aprobado <eoln>
E Lopez Desaprobado <eoln>
<eof>
```

#### Ejercicio 2:

Un comercio almacena la información de los productos que vende en un archivo de texto, **Productos.txt**, uno por línea. De cada uno guarda: **código** (entero), **stock real** (entero), **stock mínimo** (entero) y **descripción** (caracteres).

En otro archivo de texto, **Proveedores.dat**, almacena pares de números (código del producto – código del proveedor), indicando qué proveedor vende cada artículo.

Escriba un programa, que a partir de información contenida en los archivos, genere otro archivo de texto, **Compras.txt**, con el código del producto, código del proveedor y descripción de todos los artículos que están por debajo del mínimo stock.

*Por ejemplo:*

Productos.txt	Proveedores.txt	Compras.txt
100 30 15 arroz <eoln>	105 22 <eoln>	101 56 porotos <eoln>
101 20 25 porotos <eoln>	102 56 <eoln>	102 56 lentejas <eoln>
102 10 15 lentejas <eoln>	100 343 <eoln>	104 45 vinagre <eoln>
103 40 30 sal gruesa <eoln>	101 56 <eoln>	....
104 23 30 vinagre <eoln>	103 345 <eoln>	<eof>
105 22 15 aceite <eoln>	104 45 <eoln>	
...	...	
<eof>	<eof>	



**Ejercicio 3:**

Una aerolínea internacional permite realizar reservas para sus vuelos. En el archivo de números enteros **Reservas.dat** se guarda la información del pasaje reservado, la cual consiste del documento del pasajero y a continuación la fila y asiento reservado. El archivo está ordenado por número de documento.

Un ejemplo del archivo *Reservas.dat* sería:

<b>12654</b>	14	6	<b>15670</b>	15	12	<b>20300</b>	5	27	...	<eof>
--------------	----	---	--------------	----	----	--------------	---	----	-----	-------

El archivo indica que el pasajero con documento **12654** reservó la fila 14 asiento 6, el de documento **15670** reservó la fila 15 asiento 8, etc.

La aerolínea también cuenta con un archivo de texto llamado **Datos.txt** conteniendo el número de documento, apellido y nombre de los pasajeros que han hecho alguna reserva.

Un ejemplo del archivo *Datos.txt* sería:

<b>12654</b>	Perez Ana	<eoln>
<b>25412</b>	Huerta María	<eoln>
<b>20300</b>	Juarez Juan	<eoln>
...		
<b>15670</b>	Dural Alejandro	<eof>

Escriba un programa en Pascal que permita ingresar un número de documento, e indicar por pantalla el nombre del pasajero, la fila y el asiento reservados por el mismo.

Por ejemplo:

**Ingrese el DNI del pasajero:** 15670 <Enter>  
 Dural, Alejandro      Fila 15 Asiento 12

**Ejercicio 4:**

Utilizando los archivos del ejercicio anterior, escriba un programa Pascal para mostrar por pantalla los datos del vuelo, esto es, fila y asiento de cada uno de los pasajeros. *¿Se podrán utilizar los algoritmos del ejercicio anterior?*

**DATOS DEL VUELO:**

Perez, Ana      Fila 14 Asiento 16  
 Huerta, María      Fila 10 Asiento 11  
 Juarez, Juan      Fila 5 Asiento 27

...

**Ejercicio 5:**

Se dice que  $M$  es el número maximal para  $N$ , si  $M$  es **el mayor número que puede formarse usando los dígitos de  $N$** . Ejemplos: Si  $N=125345$ , el número maximal  $M$  es 554321; si  $N=2756$ ,  $M$  es 7652.

- Realice una función que determine si un número es maximal.
- Realice una función que dado un número retorne el maximal que puede formarse utilizando sus dígitos.
- Escriba un programa en Pascal que lea dos números naturales  $a$  y  $b$  y muestre por pantalla todos los números  $Num$  comprendidos entre  $a$  y  $b$  que verifiquen la propiedad de ser maximal.

Por ejemplo: para  $a = 320$  y  $b = 332$ , el programa deberá mostrar por pantalla:  
 Los números maximales entre 320 y 332 son: 320, 321, 322, 330,331, 332



**Ejercicio 6:**

Un profesor almacenó los datos de los alumnos de su materia en un archivo **alumnos.txt**. En cada línea guardó el número de inscripción del alumno y sus tres notas finales (oral, escrito y trabajos prácticos). El archivo está ordenado por número de inscripción.

En otro archivo, ordenado alfabéticamente por apellido, guarda por línea, número de inscripción, apellido y nombre de cada uno.

Desea escribir un programa para generar un archivo **Promoción.txt** con los apellidos y nombres de los alumnos que promocionan la materia, esto es, alumnos que el promedio de las tres notas supere los 7 puntos. El archivo debe quedar ordenado alfabéticamente.

**Ejercicio 7: (Parcial 2015)**

Considere una empresa que vende productos tanto en sus sucursales como por una aplicación web. Cada vez que se vende un producto en una sucursal, el código del producto (un número entero positivo) se almacena en el archivo **SUC.dat** y cada vez que un producto se vende por internet, su código se almacena en otro archivo **WEB.dat**. Además la empresa tiene un tercer archivo de enteros positivos, llamado **productos.dat**, con todos los códigos de productos que vende (en este archivo no hay elementos repetidos).

Se debe escribir un programa en Pascal que, utilizando los archivos indicados antes, permita realizar estas tareas:

1. Ingresar un código de producto y mostrar por pantalla por cuál medio se vendió más, si el producto no se vendió por ningún medio o se vendió igual cantidad debe indicarse.
2. Utilizando los códigos del archivo **productos.dat** generar un reporte por pantalla de los productos que no se vendieron por ningún medio.

*Ejemplo: Si los archivos son*

<b>Productos.dat:</b>	100, 101, 102, 103, 104, 105, 106
<b>SUC.dat:</b>	103, 100, 101, 104, 101, 104, 105, 103, 104
<b>WEB.dat:</b>	103, 104, 100, 105, 103, 105, 100

(1) Mejor Medio (2) Productos sin venta (3) Salir > 1 (Se elige la opción 1)

Ingrese código de producto: 100

**El producto se vendió más por WEB.**

1) Mejor Medio (2) Productos sin venta (3) Salir > 2 (Se elige la opción 2)

**Productos que no fueron vendidos: 102 106**

**Ejercicio 8:**

En el archivo **Texto.txt**, se almacenó un texto formado por oraciones, una oración por línea. Una oración está formada por palabras separadas por un único espacio. La oración termina con un punto.

Escribir un programa que lea el archivo **Texto.txt** y genere otro archivo informando cuántas palabras y cuántos caracteres tiene cada oración del archivo original.

El formato del nuevo archivo será:

*Oración 1: ... palabras, ... caracteres.*

*Oración 2: ... palabras, ... caracteres.*



**Ejercicio 9 (Parcial 2017):**

La biblioteca del barrio mantiene la información de sus libros en un archivo de texto, **Libros.txt**. Cada línea del archivo almacena: el código del libro (un número entero) y una secuencia de caracteres (el título y el autor del libro).

Los datos de los libros prestados se guardan en otro archivo de texto, **Prestados.txt**, una línea por cada préstamo, conteniendo el código del libro prestado y el número de socio a quién fue prestado (ambos números enteros).

Un tercer archivo de texto, **Morosos.txt**, contiene los números de todos los socios que están atrasados en las entregas de los libros (un número entero por línea).

**Dividiendo el problema en sub-problemas**, escriba un programa en Pascal para que ingresando el código de un libro y el número de un socio, indique por pantalla si se autoriza el préstamo: para esto el código del libro debe estar en **libros.txt**, el libro **no** debe estar prestado y el socio **no** debe figurar en el archivo de morosos. Si el préstamo es posible, se debe actualizar el archivo **Prestados.txt** agregando el código del libro y el número de socio en una línea al final del archivo. Si el préstamo no es posible, debe indicar por qué: Libro inexistente (no está en Libros.txt), Libro Prestado o Socio Moroso.

**Nota:** La solución será correcta, solo si el problema es dividido en sub-problemas y cada sub-problema está implementado con una función o procedimiento que son utilizados por el programa para resolver el problema planteado.

Por ejemplo, suponiendo que los archivos contienen la siguiente información:

Archivos	Libros.txt	Prestados.txt	Morosos.txt
<b>Contenido</b>	234 Matemática I Roca, Juan	154 46787	29365
	154 Matemática II Rena, Tomás	292 29365	28009
	117 La Guerra Chevro, Ana	170 28009	
	292 El Jardín García, Horacio	296 26987	
	654 Viajar Citto, Fabio		
	170 Física I Lopez, María		

Por pantalla se deberá mostrar (respetar el formato):

- |  |                            |
|--|----------------------------|
| <b>Ingrese código del libro y número de socio:</b> 100 26987 | <b>LIBRO INEXISTENTE</b>   |
| <b>Ingrese código del libro y número de socio:</b> 154 24530 | <b>LIBRO PRESTADO</b>      |
| <b>Ingrese código del libro y número de socio:</b> 117 29365 | <b>SOCIO MOROSO</b>        |
| <b>Ingrese código del libro y número de socio:</b> 117 26987 | <b>PRESTAMO AUTORIZADO</b> |

*(Este ejercicio podrá ser entregado al asistente para su corrección).*