

## Strings: Ejercicios

1.

[25 %] En el básquetbol existen tres diferentes tipos de anotaciones:

- el tiro libre (L), que vale un punto,
- el doble (D), que vale dos puntos, y
- el triple (T), que vale tres puntos.

Un partido de básquetbol está dividido en varios períodos.

Usted debe escribir un programa que reciba como entrada una única línea, que contenga todas las anotaciones realizadas por un equipo de básquetbol durante un partido. Las anotaciones de períodos distintos deben ir separadas por un espacio. Como salida, debe mostrar la cantidad de puntos obtenidos en cada período y los puntos totales, siguiendo el formato del ejemplo.

```
Anotaciones: DDTDLLDD DDLDT TDTLLD DDDDD
15 puntos en el periodo 1
10 puntos en el periodo 2
12 puntos en el periodo 3
10 puntos en el periodo 4
Total: 47 puntos
```

2. Dado un string con el siguiente formato, pero del que desconocemos la cantidad de asignaturas: "Progra=78;Mate=83;Física=68;Química=65". Escriba un programa que lea el string como entrada y calcule el promedio de calificaciones, indicando además la materia con mejor promedio. En caso de empate puede mostrar cualquiera de las que empatan.

3. Fechas mágicas: Una fecha mágica es una fecha en la que el día multiplicado por el mes es igual a los últimos dos dígitos del año. Por ejemplo, el 10 de junio de 1960 es una fecha mágica, pues junio es el mes 6, y al multiplicarlo por 10 el resultado coincide con el año: 60. Escriba un programa que determine si una fecha es mágica o no. La fecha a analizar es un string en formato "mes día, año", con un espacio separando el mes y el día, y una coma y un espacio separando el año. Por ejemplo, "Junio 10, 1960".

4. Una cadena de ADN es válida si está compuesta únicamente por las bases Adenina (A), Citosina (C), Guanina (G) o Timina (T). Escriba un programa para validar una cadena de ADN. Suponga que la cadena está compuesta por múltiples grupos de 4 letras separados por guiones. Luego, escriba un programa que lea n cadenas, y vaya señalando las que son válidas. Al finalizar debe decir cuántas fueron válidas y cuántas no.

**5.** Escriba un programa que construya un string con las letras que coinciden en dos strings ingresados como entrada. Por ejemplo, “amorosos” y “amortiza” coinciden en: “amor”; por otra parte, “conformidad” y “contorno” coinciden en “conor”. Observe que los strings pueden tener distintos largos.

**6.** Escriba un programa que lea como entrada dos strings a comparar y un nivel de tolerancia que es un número entero no negativo (cero o más). El programa debe indicar si los strings son iguales ignorando diferencias hasta la cantidad de tolerancia indicada. Por ejemplo, “perro” y “perXo” son iguales para tolerancia 1, pero son distintos para tolerancia 0.

**7.** Escriba un programa que determine si una contraseña es suficientemente segura. Una contraseña se considera suficientemente segura si contiene al menos una letra mayúscula, al menos una minúscula, al menos un dígito, al menos un caracter de puntuación (punto, coma, punto y coma o dos puntos), y si tiene al menos longitud 8.

**8.** Escriba un programa que indique la cantidad de veces que un string aparece dentro de otro. Puede suponer que el primero tiene menor longitud que el segundo.

**9.** Escriba un programa que lea como entrada un texto y una palabra, y construya otro string similar al texto original, pero que no contenga todas las ocurrencias de la palabra ingresada.

**Nota:** En AULA encontrará este mismo documento, pero con una página adicional con un ejercicio de desafío.

## Desafío

Escriba un programa que lea como entrada un *string* que contiene un número romano válido, e imprima como salida un entero con la representación decimal del número romano ingresado.

El valor de los dígitos romanos se muestra en la siguiente tabla:

I	V	X	L	C	D	M
1	5	10	50	100	500	1000

Para la conversión, tome en cuenta las siguientes reglas:

- Si a la derecha de una cifra romana se escribe otra igual o menor, su valor se suma a la anterior. Por ejemplo: XX (10+10=20) o XV (10+5=15).
- Si entre dos cifras romanas cualesquiera existe otra menor (I, X ó C), se restará su valor a la siguiente. Por ejemplo: IX (10-1=9) o CM (1000-100=900).
- En ningún número se puede poner una misma letra más de tres veces seguidas.

Ejemplos:

Ingrese un número romano: CLXIII  
163

Ingrese un número romano: CXLIX  
149

Explicación:

Romano: CLXIII	C	L	X	I	I	I
Decimal: 163	+100	+50	+10	+1	+1	+1

Romano: CXLIX	C	X	L	I	X
Decimal: 149	+100	-10	+50	-1	+10