

# Programación 1

Analista Programador - Analista en Tecnologías de la Información

Universidad ORT Uruguay

Strings

1 ★☆☆



Solicitar un texto en pantalla y luego mostrarlo en forma invertida.



Ejemplo, si contiene: 'hola mundo', obtener 'odnum aloh'

2 ★☆☆



Solicitar un texto y una letra e indicar cuántas veces aparece esa letra en el texto Ingresado.

3 ★☆☆



Solicitar un texto e indicar cuantas vocales presenta.

4 ★☆☆



Ingresar un texto y al presionar un botón convertirlo a minúsculas y mostrarlo en un párrafo.

5 ★☆☆



Ingresar un texto y al presionar un botón convertirlo a mayúsculas y mostrarlo en un párrafo.

6 ★☆☆



Ingresar dos textos y verificar si el segundo es una subcadena del primero, es decir si está comprendido completamente dentro del primero.

Por ejemplo: "tom" es una subcadena de "automóvil descapotable".

## 7 ★☆☆



Dado un texto contar la cantidad de palabras que contiene. Para separar palabras en esta etapa consideraremos como separador de palabras únicamente los espacios en blanco.

## 8 ★☆☆



Dado un texto indicar si comienza y termina con la misma letra.

## 9 ★☆☆



Sabiendo que la primera letra de una matrícula indica el departamento al que pertenece el vehículo. Ingresar una matrícula (formato: AAA-1111) e informar a que departamento pertenece ese vehículo.

A - Canelones  
B - Maldonado  
C - Rocha  
D - Treinta y Tres  
E - Cerro Largo  
F - Rivera  
G - Artigas  
H - Salto  
I - Paysandú  
J - Río Negro  
K - Soriano  
L - Colonia  
M - San José  
N - Flores  
O - Florida  
P - Lavalleja  
Q - Durazno  
R - Tacuarembó  
S - Montevideo

## 10 ★☆☆



Dado un texto y una letra mostrar el texto ingresado, reemplazando la letra indicada por "\*". Por ejemplo si se ingresa el texto "tres tristes tigres comen trigo en un trigal" y como letra se ingresa "r", lo que se debe mostrar es: t\*es t\*istes tig\*es comen t\*igo en un t\*igal

## 11 ★★★



Ingresar un texto y al presionar un botón convertir todo el texto a minúsculas y solo la primera letra de la primera palabra a mayúsculas.

## 12 ★★★



Hacer un programa que ingrese una cadena de caracteres y determine el número de mayúsculas y el número de minúsculas.

Se deberán eliminar los espacios en blanco antes de hacer el cálculo y se puede considerar cualquier caracter que no sea a-z o A-Z como mayúscula o minúscula.

En caso de generar una solución más compleja se puede consultar la siguiente tabla.

1	26 →	51 3	76 L	101 e	126 ~	151 —	176 °	201 É	226 â	251 û
2 7	27 ←	52 4	77 M	102 f	127 Ì	152 °	177 ±	202 Ê	227 ã	252 ü
3 8	28	53 5	78 N	103 g	128 €	153 ™	178 ²	203 Ë	228 ä	253 ý
4 9	29	54 6	79 O	104 h	129	154 Š	179 ³	204 Ì	229 å	254 þ
5 0	30	55 7	80 P	105 i	130 ,	155 ›	180 ´	205 Í	230 æ	255 ÿ
6 1	31	56 8	81 Q	106 j	131 f	156 œ	181 μ	206 Î	231 ç	
7 2	32	57 9	82 R	107 k	132 „	157	182 ¶	207 Ï	232 è	
8 3	33 !	58 :	83 S	108 l	133 ...	158 ž	183 ·	208 Ð	233 é	
9 4	34 "	59 ;	84 T	109 m	134 †	159 Ÿ	184 ˆ	209 Ñ	234 ê	
10 5	35 #	60 <	85 U	110 n	135 ‡	160	185 ˜	210 Ò	235 ë	
11 6	36 \$	61 =	86 V	111 o	136 ×	161 ¡	186 ø	211 Ó	236 ì	
12 7	37 %	62 >	87 W	112 p	137 ‰	162 ¢	187 »	212 Ô	237 í	
13 8	38 &	63 ?	88 X	113 q	138 Š	163 £	188 ¼	213 Õ	238 î	
14 9	39 '	64 @	89 Y	114 r	139 ‹	164 ¤	189 ½	214 Ö	239 ï	
15 0	40 (	65 A	90 Z	115 s	140 Œ	165 ¥	190 ¾	215 ×	240 ð	
16 1	41 )	66 B	91 [	116 t	141	166 ¤	191 ¿	216 Ø	241 ñ	
17 2	42 *	67 C	92 \	117 u	142 Ž	167 §	192 À	217 Ù	242 ò	
18 3	43 +	68 D	93 ]	118 v	143	168	193 Á	218 Ú	243 ó	
19 4	44 ,	69 E	94 ^	119 w	144	169 ©	194 Â	219 Û	244 ô	
20 5	45 -	70 F	95 _	120 x	145 ´	170 ª	195 Ã	220 Ü	245 õ	
21 6	46 .	71 G	96 `	121 y	146 ´	171 «	196 Ä	221 Ý	246 ö	
22 7	47 /	72 H	97 a	122 z	147 ¨	172 ¬	197 Å	222 Þ	247 ÷	
23 8	48 0	73 I	98 b	123 {	148 ¨	173 ¯	198 Æ	223 ß	248 ø	
24 9	49 1	74 J	99 c	124	149 •	174 ®	199 Ç	224 à	249 ù	
25 0	50 2	75 K	100 d	125 }	150 —	175 ¯	200 È	225 á	250 ú	



Con la función `charCodeAt` se puede limitar la verificación a caracteres de la a a la z en mayúscula o minúscula.

## 13 ★★☆☆



a) Dado un texto Indicar si es un palíndromo. Tener en cuenta que no se tienen en cuenta los espacios en blanco.

Ej: "Amargor pleno con el programa" , "Ana lleva al oso la avellana",

b) También se consideran palíndromos los textos en los que no coinciden los acentos o las puntuaciones.

Ej: "Dábale arroz a la zorra el abad", "Saco sal, ayo, del yodo y le doy a las ocas"

**i** Un palíndromo (del griego palin dromein, volver a ir hacia atrás) es una palabra, número o frase que se lee igual hacia adelante que hacia atrás. Si se trata de un número, se llama capicúa.

## 14 ★★☆☆



Escribir una función que reciba como parámetro una palabra y devuelva la cantidad de veces que aparece la primera letra de esa palabra en toda la palabra.

Ej.: Para la palabra "aumentativo" se debe devolver un 2.

Ej.: Para la palabra "comunicaciones" se debe devolver un 3.

## 15 ★★★★★



Crear una función que reciba como parámetro un texto y devuelva la posición en la que se encuentra la penúltima ocurrencia de la letra "a" dentro de ese texto. En caso de que haya una sola "a" o no haya ninguna, devolver -1.

Ej. Para la palabra autoabastecimiento se debe devolver un 4 que es la posición de la penúltima "a".

## 16 ★★★★★



Ingresar una cédula de identidad uruguaya en un campo de texto con formato 1.111.111-1.

Utilizando el algoritmo de cálculo del dígito verificador (el posterior a la barra), verificar si la ingresada es una cédula válida.

Se toman los primeros siete números y se multiplican cada uno (respetando el lugar, es decir: primero x primero, segundo por segundo) por 2987634.

Si la cédula es 1.234.567 la cuenta quedaría así:

$$1 \times 2 + 2 \times 9 + 3 \times 8 + 4 \times 7 + 5 \times 6 + 6 \times 3 + 7 \times 4 = 148.$$

Se toma el resultado y se le busca el número mayor que termina en 0 y se le resta. Ej:  $150 - 148 = 2$ .

El número 2 es el dígito verificador para la cédula 1.234.567

**Casos de prueba de cédulas válidas:**

394.281-3 | 3.966.136-4 | 3.225.140-7 | 5.069.005-9 | 4.946.342-9 | 600.346-2



Crear un campo de texto para ingresar un valor y verificar si valida con el algoritmo de Luhn.

El **algoritmo de Luhn** o **fórmula de Luhn**, también conocida como "**algoritmo de módulo 10**", es una fórmula de **suma de verificación**, utilizada para validar una diversidad de números de identificación; como **números de tarjetas de crédito**, **números IMEI**, etc.

La fórmula verifica un número contra su dígito de chequeo incluido, el cual es usualmente agregado a un número de cuenta parcial para generar el número de cuenta completo. Este número de cuenta debe pasar la siguiente prueba:

- 1. A partir del dígito de chequeo incluido, el cual está a la derecha de todo el número, ir de derecha a izquierda duplicando cada segundo dígito.
- 2. Sumar los dígitos del resultado: (ejemplo:  $10 = 1 + 0 = 1$ ,  $14 = 1 + 4 = 5$ ) juntos con los dígitos sin duplicar del número original.
- 3. Si el total de **módulo 10** es igual a 0 (si el total termina en cero), entonces el número es válido de acuerdo con la fórmula Luhn, de lo contrario no es válido.

Supongamos un ejemplo de un número de cuenta "7992739871", que contará con un dígito de control adicional, por lo que es de la forma 7992739871x:

Dígitos del número de cuenta	7	9	9	2	7	3	9	8	7	1	x
Duplicar cada siguiente	7	18	9	4	7	6	9	16	7	2	x
Sumar los dígitos	7	9	9	4	7	6	9	7	7	2	=67

El dígito de chequeo (x) se obtiene entonces de  $(67 * 9 \bmod 10)$ . En términos sencillos:

- 1. Calcular la suma de los dígitos (67).
- 2. Multiplicar por 9 (603).
- 3. Tomar el último dígito (3).
- 4. El resultado es el dígito de chequeo.

Casos de prueba

Visa	Mastercard	Discover	American Express
<ul style="list-style-type: none"><li>• 4485746493384014</li><li>• 4539371567070872</li><li>• 4539625308478250</li><li>• 4916103567334187</li></ul>	<ul style="list-style-type: none"><li>• 5223450370829605</li><li>• 5506255981334939</li><li>• 5183026692100376</li><li>• 5508191347495098</li><li>• 5243130311850979</li></ul>	<ul style="list-style-type: none"><li>• 6011644263302639</li><li>• 6011518345624527</li><li>• 6011319554023976</li><li>• 6011573093875086</li><li>• 6011956548219903</li></ul>	<ul style="list-style-type: none"><li>• 340335960322765</li><li>• 378182614820157</li><li>• 342437599153724</li><li>• 346840561012917</li><li>• 378344449580014</li></ul>



## 18 ★★★★★



Las matriculas de autos en Uruguay tienen un formato de tres letras y cuatro números o una letra y seis números.

- a) Dado un ingreso de matrícula verificar si es una matrícula válida, en caso que no, informarlo.
- b) Mejorar la solución anterior para eliminar todos los caracteres que no sean letras o números antes de evaluar la matrícula.

