

Facultad de Ingeniería UABC
Tecnologías de la Programación
Práctica 2: Cadenas de Caracteres

EJERCICIO 1:

El término “l33t Speak” (pronunciado “lit”) se refiere a un lenguaje o un sistema de notación ampliamente usada por hackers. Esta notación es única porque no puede ser escrita a mano o hablada. Es una notación basada en la internet que recae en el teclado. Un ejemplo importante es la frase “I am elite”, la cual se convertiría en “1 4m 3l1t3”, los hackers escriben muy mal, es por eso que la frase fue mal convertida a “1 4m 3l33t” después se convirtió en “1 4m l33t” la cual le dio a l33t Speak su nombre actual.

Como cualquier otro lenguaje o sistema de notación, l33t Speak tiene sus reglas de gramática. Las reglas básicas (del nivel básico de l33t) para reemplazar letras con dígitos u otro carácter son las siguientes:

Letra	Dígito
A/a	4
E/e	3
I/i	1
O/o	0
T/t	7
S/s	5

En el caso de los números, deberán ser convertidos a su notación escrita, por ejemplo:

“4 perros ladran” → “Cu47r0 p3rr05 l4dr4n”

Ejemplo de un mensaje en “l33t” speak:

“Cada día sabemos más y entendemos menos”

→ “c4d4 d14 54b3m05 m45 y 3n73nd3m05 m3n05”

Hacer un programa en C que ofrezca un menú de opciones como el que se muestra a continuación:

Conversión a L33t

1. Leer una cadena
2. Convertir cadena a L33t
3. Imprimir cadena
4. Terminar programa

1. **Leer cadena:** Esta opción permitirá leer una cadena de caracteres con una longitud máxima de 50. La cadena podrá permitir únicamente letras (mayúsculas y minúsculas) y números.
2. **Convertir cadena a L33t:** Esta opción permitirá realizar la conversión de la cadena ingresada anteriormente a su representación en L33t Speak sin mostrar la conversión, generar la cadena resultante y ésta deberá estar solamente escrita con letras minúsculas. Esta opción no podrá ser utilizada sin antes ingresar una cadena con la opción 1.
3. **Imprimir cadena en L33t:** Esta opción permitirá imprimir la cadena original y su representación en L33t. Esta opción no podrá ser utilizada sin antes ingresar una cadena y convertirla.

Facultad de Ingeniería UABC
Tecnologías de la Programación
Práctica 2: Cadenas de Caracteres

Facturación de venta

Hacer un programa en C que lea dos cadenas de caracteres con una longitud máxima de 100 caracteres y que contenga los siguientes datos:

Cadena 1:

Nombre completo, dirección, correo electrónico.

Cadena 2:

#cantidad de producto, descripción del producto, \$precio unitario, % de Iva

Esta cadena puede contener más de 1 producto lo que se identificará con un “#”.

El programa deberá validar que los datos estén correctos y que se desglose como se muestra en el siguiente ejemplo:

Cadenas leídas desde el teclado:

Cadena No.1: López Martínez Carla, Calle rio presidio #1914 cp 24874, martines.carla@hotmail.com

Cadena No.2: #2, Lámpara LED, \$500, % 16, #5, Ventiladores, \$1000, % 17.

Salida generada por el programa:

Nombre: López Martínez Carla

Dirección: Calle rio presidio #1914

Código Postal: 24874

Correo electrónico: martinez.carla@hotmail.com

Cantidad	Descripcion	Costo
2	Lampara LED	1160
5	Ventilador	5850.

Se deben considerar las características que a continuación se enlistan:

- **Nombre:** Permitirá solamente letras, letras acentuadas, diéresis y apóstrofes.
- **Dirección:** Permitirá solamente letras, letras acentuadas, y números.
- **Correo electrónico:** se deberá validar que la cadena sea válida, esto es, que incluya caracteres válidos para un correo electrónico (letras, números, guion bajo, guion medio, punto), y deberá incluir obligatoriamente un solo carácter de arroba. Cualquier error deberá ser notificado.
- **Cantidad de producto:** se deberá validar que sea un número entero y que contenga el "#".
- **Descripción del producto:** Permitirá solamente letras, números, letras acentuadas, diéresis y apóstrofes.
- **Costo:** El costo será calculado a partir de la suma del (precio unitario * cantidad de producto) + el porcentaje de IVA. Así que se deberá validar que la cadena cuenta con el precio unitario identificado después de un \$ y el IVA después de un %.