

# ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

SEGUNDA EVALUACIÓN  
FUNDAMENTOS DE PROGRAMACIÓN  
II TÉRMINO 2010 – 2011

NOMBRE: \_\_\_\_\_ PARALELO: \_\_\_\_\_

## TEMA 1 (20 puntos)

Escriba el **procedimiento recursivo eliminarEspacios(char \*s1, char \*s2)**, el cual recibe la cadena s1 y coloca en s2 la cadena sin los espacios en blanco. *Nota:* Asuma que s2 inicialmente está vacía. *Por ejemplo:*

s1: salu2 para todos                      s2: salu2paratodos  
s1: vamos a la playa! oh oh oh        s2: vamosalplaya!ohohoh

## TEMA 2 (35 puntos)

Scrabble es un juego de mesa en el cual cada jugador intenta obtener puntos mediante la construcción de palabras. El juego se desarrolla sobre un tablero de 9x9, el cual tiene casillas de premiación que multiplican el número de puntos concedidos.

Implemente la función:

**bool colocarPalabra (char T [9][9], int f, int c, orientacion o, char \*palabra);**

La función coloca la *palabra* que recibe como parámetro en el tablero T, iniciando en la posición dada por la fila *f* y la columna *c* sólo si cumple con lo siguiente:

- La palabra no se sale del tablero
- La palabra está en la lista de palabras del diccionario
- La palabra usa al menos una letra del tablero excepto cuando el tablero está vacío. En este caso la primera letra de la palabra debe colocarse en el centro del tablero.
- Cada casilla donde se coloque una letra está vacía ó coincide con la letra colocada en el tablero.

La palabra se coloca en mayúsculas y según la orientación *o*, el cual corresponde al tipo:

**enum orientación {VERTICAL, HORIZONTAL};**

La función *colocarPalabra* retorna *true* si se colocó con éxito sino retorna *false*.

**NOTA:** Asuma que Ud. posee las funciones *estaVacio* y *esPalabra* cuyos prototipos son:

**bool estaVacio (char T[9][9]);**

//Retorna *true* si el tablero está vacío sino retorna *false*.

**bool esPalabra(char \*palabra);**

//Retorna *true* si la palabra que recibe como parámetro se  
//encuentra en el diccionario estándar de juego  
//sino retorna *false*.

								A	
								V	
								I	
								O	
				L	E	O	N		
				I					
				B					
				R					
			P	O	E	M	A		

### TEMA 3 (45 puntos)

Una empresa desea crear un sistema de diccionarios para niños con los idiomas: español, inglés y francés. El sistema debe mostrar un menú con las siguientes opciones:

- 1 espaniol-ingles
- 2 ingles-frances
- 3 Buscar palabras
- 4 Salir

Se ha definido la siguiente estructura:

```
typedef struct {
    int id;
    char esp[MAX];
    char eng[MAX];
    char fr[MAX];
}idiomaT;
```

diccionario.dat

```
1,casa,2,house,3,maison
2,table,3,table,1,mesa
3,fille,2,daughter,1,hija
2,she,1,ella,3,elle
...
```

A partir del archivo `diccionario.dat` y la estructura `idiomaT` implemente:

- a) El procedimiento **cargarPalabras(FILE \*pf, idiomaT P[ ], int \*tam)** recibe un puntero al archivo "diccionario.dat", carga su contenido en un arreglo de estructuras y actualiza la variable `tam` con el número de registros leídos. En el campo "id" se guarda el número de registro (fila del archivo). El archivo contiene una lista de palabras, en cada línea hay tres palabras con un número que representa el idioma en que está escrita: (1)Español, (2)Ingles y (3)Francés.
- b) El procedimiento **generarDic(FILE \*pDic, idiomaT P[ ], int tam, int tipo)** que recibe el tipo de diccionario que se desea generar: (1)Español-Inglés o (2)Ingles-Francés. El procedimiento `generarDic` coloca en el archivo referenciado por `pDic` la palabra en el idioma fuente y en el idioma destino. Por ejemplo: En el caso de elegir Español-Inglés `pDic` referencia a un archivo como el siguiente:

esp-eng.dat

```
casa: house
mesa: table
hija: daugther
ella: she
```

- c) El procedimiento **buscarPalabras(idiomaT P[ ], int tam, char \* subcadena)** recibe el arreglo `P` de tamaño `tam` y `subcadena`. Este procedimiento mostrará en pantalla todas las palabras que contengan la `subcadena` enviada indicando el idioma en el que se encuentran. Por ejemplo:

```
Subcadena: cha
chantilly (esp)
enchanted (eng)
chateau (fr)
lucha (esp)
```

- d) El **programa principal** que muestre el menú del sistema y ejecute las opciones seleccionadas por el usuario. Use los procedimientos implementados en los literales anteriores.