

# Auxiliar N°1 - CC3301

Alonso González Ulloa

April 16, 2010

## Problema 1

Implemente una lista enlazada en C construyendo el tipo LISTA e implementado la siguiente interfaz:

```
/* * Lista simplemente enlazada */

/* agrega el elto p al comienzo de la lista l */
LISTA *cons_l(void *p, LISTA *l);

/* Retorna el elemento más nuevo de la lista y
 * se posiciona al comienzo */
void *first_l(LISTA *l);

/* Retorna el elemento siguiente de la lista */
void *next_l(LISTA *l);

/* Libera la lista completa */
void free_l(LISTA *l);
```

## Problema 2

Definimos un grafo como el par  $G = (V, E)$  con  $V = \{v_1, \dots, v_n\}$  un conjunto finito de vértices y  $E \subseteq V^2$  un conjunto de aristas. Implemente una grafo en C construyendo el tipo GRAFO e implementado la siguiente interfaz:

```
/* Grafo dirigido */

/* Crea un nuevo nodo del grafo */
nodo *crear_nodo(void *p, LISTA *ady);

/* Libera la memoria del grafo */
void liberar_nodo(nodo *n);
```

### Problema 3

Definimos un Automa Finito Determinista (AFD) como la 5-tupla

$$A = (Q, \Sigma, \delta, s, F)$$

Donde:

- $Q$  es un conjunto finito de estados.
- $\Sigma$  es un alfabeto finito.
- $\delta$  es la función de transición,  $\delta : Q \times \Sigma \rightarrow Q$ .
- $s$  es el estado inicial.
- $F \subseteq Q$  es el conjunto de estado:s finales.

Extendemos la función  $\delta$ , para obtener  $\delta' : Q \times \Sigma^* \rightarrow Q$ , de la siguiente forma:

$$\delta'(q, \omega) = \begin{cases} q & |\omega| = 0 \\ \delta(\delta'(q, \omega_1 \dots \omega_{|\omega|-1}), \omega_{|\omega|}) & |\omega| \geq 1 \end{cases}$$

Finalmente decimos que  $A$  acepta  $\omega$  ssi  $\delta'(s, \omega) \in F$ . (Mejor mirar un monito)

Usando los resultados de la pregunta 2 suponga lo necesario para que un grafo se comporte como un AFD y programe en C la función  $\delta'$ .