

Algoritmos y Programación I (95.11) – Curso Essaya – 4^{to} parcialito, 2^{do} recuperatorio – 12/12/2019

Resolver los siguientes problemas en forma clara y legible en código ISO-C99.

1. Teniendo una lista **de enteros** en la cual la lista se representa `typedef struct {struct nodo *prim;} lista_t`; y el nodo se representa como `struct nodo {struct nodo *sig; int dato;}`; implementar la primitiva

```
void lista_enteros_borrar(lista_t *l, int e);
```

que borre todas las ocurrencias del dato `e` en la lista.

2. Para un evento que duró 2 días se tienen dos vectores que cada uno de ellos contiene los DNIs de los asistentes al primero y al segundo día ordenados por número de DNI. Se quiere obtener una lista con los asistentes únicos al evento, es decir, una lista que consolide los asistentes de los dos días sin duplicados. Escribir una función:

```
int *consolidar_asistentes(const int dia1[], size_t n1, const int dia2[], size_t n2, size_t *n_asis);
```

que, de forma eficiente, consolide y devuelva un vector con los DNIs de los asistentes. Debe devolver en `n_asis` el tamaño del vector resultante. (**Nota:** No importa si sobra memoria en el vector.)

3. Escribir una función `bool esta_ordenado(float v[], size_t n)`; que utilizando recursividad indique si un vector `v` de `n` elementos se encuentra ordenado o no.

¡Suerte! :)