

CERTAMEN #2 ICI-142

Makarena Donoso <makarena.donos@gmail.com>

Wenceslao Palma <wenceslao.palma@ucv.cl>

En el caso de los ejercicios 1 y 2 considere:

```
struct nodo{
    int dato;
    struct nodo *sgte;
};

struct nodo *inicio; // puntero al inicio de la lista.
```

1. (25 ptos.) Escriba una función que determine el centro de una lista simple. Restricción: no está permitido el uso de variables de tipo entero. La solución debe considerar únicamente el uso de punteros.

struct nodo *centro(struct nodo *);

2. (20 ptos.) Escriba una función que permita desplazar el n-ésimo nodo de una lista simple m posiciones hacia adelante. La función debe retornar 1 cuando la operación se puede realizar, en caso contrario retorna 0.

int desplazarNodo(struct nodo **, int, int);

3. (15 ptos.) En un archivo de texto llamado “trabajadores.txt” se encuentran los datos rut:Nombre:edad de los trabajadores de una empresa. Considere que cada línea del archivo contiene los datos de un trabajador. En un segundo archivo llamado “familia.txt” se encuentran los datos de la familia de cada trabajador. Considere que en cada línea del archivo se encuentran los sgtes datos: rutTrabajador:nombreEsposa:edadEsposa:nombreHijo:edadHijo:.....

Los datos deben ser cargados a un sistema que enlace a los trabajadores en una lista doble, cada uno de ellos poseera una lista simple donde se almacenará la información de los integrantes de la familia.

Considere las sgtes definiciones :

```
struct trabajador{
    char rut[12];
    char nombre[20];
    int edad;
    struct trabajador *sgte;
    struct trabajador *ant;
    struct familiar *familia;
};

struct familiar{
    char rut[12];
    char nombre[20];
    struct familiar *sgte;
};

struct trabajador *inicio;
```

Implemente las funciones **void agregarTrabajador(struct trabajador **)** y **void agregarFamiliar(struct familiar **)**. Considere la existencia de las siguientes funciones:

- **struct trabajador crearTrabajador(char *, char *, int);** la cual recibe como parámetro los datos del trabajador y retorna el nodo completo.
- **struct familiar crearFamiliar(char *, char *);** la cual recibe como parámetro los datos del familiar y retorna el nodo completo.