



PRÁCTICA 2

TAD Lineales Dinámicos

VERSIÓN 2

Enunciado

La versión 2 de la práctica 2 va a ser una ampliación de la versión 1 donde vamos a poner en práctica TAD dinámicos implementados con estructuras de nodos enlazadas vistas en la teoría del tema 4.

Por tanto, a partir de la "lista de n colas" obtenida en la anterior versión, vamos a crear una lista doblemente enlazada y una lista circular, de nodos de los siguientes tipos:

```
struct TNode_ListaD {
    Paciente Datos;
    TNode_ListaD *Ant;
    TNode_ListaD *Sig;
};

struct TNode_ListaC {
    Paciente Datos;
    TNode_ListaC *Sig;
};
```

Cada vez que se atienda un paciente en las urgencias de la clínica, este paciente saldrá de la Cola de Prioridad y será insertado en una lista de pacientes que estén siendo atendidos en la clínica. La lista doblemente enlazada será para los pacientes de sexo Masculino y la lista circular para los pacientes de sexo Femenino. El sexo de los pacientes se simulará aleatoriamente con la función rand.

La función **rand** genera un número pseudo_aleatorio entre 0 y la constante **RAND_MAX** (número suficientemente grande). Al ser una generación de números pseudo_aleatoria (basada en una fórmula que parte de una determinada semilla, que si no se modifica será siempre la misma) cada vez que ejecutemos nuestro programa, la secuencia de números "aleatorios" será siempre igual. Para evitar este inconveniente existe la función **srand** que nos permite establecer una semilla distinta, para la generación de números aleatorios, cada vez que realicemos una ejecución del programa, para lo cual haremos uso de otra función llamada **time** (contenida en la librería **time.h**). Esta función será llamada al principio del programa y una sola vez. La forma de utilizar estas funciones será la siguiente:

```
/* Al comenzar el programa pondremos la siguiente instrucción, para inicializar el generador
de números aleatorios */
srand((unsigned) time(NULL));

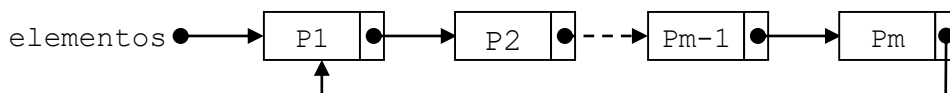
/* Para obtener un número aleatorio entre 1 y 100, y almacenarlo en una variable entera
llamada p.e. N haremos lo siguiente */
N = (rand()%100)+1;
```

Si el valor aleatorio obtenido es par el paciente será Masculino y tendremos que insertarlo en una lista doblemente enlazada de pacientes, y si el valor aleatorio obtenido es impar, el paciente será Femenino y se insertará en una lista circular.

- **Lista doblemente enlazada de pacientes masculinos.**



- **Lista circular de pacientes femeninos:**



Donde P1 a Pn son pacientes masculinos que han sido desencolados de la Cola de Prioridad y P1 a Pm son pacientes femeninos que han sido desencolados de la Cola de Prioridad.

Para poder probar el funcionamiento de las listas de pacientes, habrá que añadir una opción al menú principal de la aplicación que permita listar dichas listas.

----- Menú Principal -----

1- Cargar datos desde fichero
2- Admisión de un nuevo Paciente
3- Atender un Paciente
4- Listar Pacientes esperando para atender
5- Listar Pacientes según prioridad
6- Listar Pacientes Atendidos
7- Salir

Opción 6. Preguntará qué pacientes se quieren mostrar: los pacientes masculinos, los femeninos o todos, y los mostrará por pantalla.

Es obligatorio el uso de diseño modular.

Fecha de finalización

La práctica completa (versión 1 y 2) deberá estar finalizada antes de la realización de la segunda prueba de modificación de prácticas (última sesión práctica de la asignatura), fecha que será comunicada por los profesores de la asignatura y publicada en el portal de apoyo de la docencia semipresencial de la UHU (aulasvirtuales.uhu.es).