

ICI-142 Fundamentos de Programación

Tarea #6

Wenceslao Palma <wenceslao.palma@ucv.cl>

En Ciencias de la Computación, una lista enlazada es una de las estructuras de datos fundamentales y puede ser usada para implementar otras estructuras de datos. Una lista enlazada consiste en una secuencia de nodos, en los que se guardan campos de datos arbitrarios y referencias a otros nodos. Los nodos de una lista enlazada se construyen utilizando un tipo de dato autoreferenciado ya que contienen un puntero o enlace a otro nodo del mismo tipo. Las listas enlazadas permiten inserciones y eliminación de nodos en cualquier punto de la lista en tiempo constante (suponiendo que dicho punto está previamente identificado o localizado), pero no permiten un acceso aleatorio.

En la presente tarea consideraremos el uso de listas simplemente enlazadas para buscar un elemento dentro de una secuencia ordenada de números. Una técnica para aquello consiste en la utilización de punteros de salto los cuales son ubicados cada cierto número de nodos. Los punteros de salto hacen referencia a otro nodo en el cual hay un puntero de salto, experimentalmente se ha mostrado que una buena decisión es ubicar los punteros de salto cada \sqrt{n} nodos. Por ejemplo, para una lista de 25 nodos los punteros de salto se ubican cada 5 nodos.

Por ejemplo, para una lista de 25 números los punteros de salto (*) se ubican de la siguiente manera.

```
13 17 23 35 37 39 45 47 50 51 56 59 60 62 64 72 75 77 80 84 86 90 94 110 114
*      *      *      *      *      *      *
```

Luego, para una operación de búsqueda se utilizan los punteros de salto de la siguiente manera: si se desea buscar el 64, ya que $64 > 13$ se salta del 13 al 37, como $64 > 37$ se salta al 50, luego al 60 y posteriormente al 75. Sin embargo, ya que $64 < 75$ se buscará su existencia en forma lineal entre el 60 y el 75.

Para el desarrollo de la tarea considere el uso de la siguiente definición de nodo:

```
typedef struct nodo{
    int valor;
    struct node *sgte;
    struct node *salto;
}nodo;

nodo *INICIO;
```

El objetivo de la presente tarea es construir, a partir de un archivo de entrada, una lista enlazada ordenada, ubicar en ella los punteros de salto y responder

una serie de consultas de búsqueda. Para el ordenamiento no se permite el uso de estructuras de datos auxiliares, se debe utilizar sólo la lista enlazada. Recomendación: ordenar en la medida que se insertan los nodos en la lista.

Entrada

La entrada se encuentra en un archivo de texto llamado **in.txt** el cual contiene un conjunto desordenado de números enteros positivos y un conjunto de consultas, todos separados por un espacio. Los números que componen el conjunto terminan cuando se encuentra el valor -1 y las consultas cuando se encuentra un -2 . Por ejemplo:

```
86 35 75 13 51 23 37
45 84 17 50 39 56 60 114
64 72 94 77
80 47 90 59 110 62 -1 64 10 50 94 -2
```

Salida

La salida del programa se debe almacenar en un archivo de texto llamado **out.txt**. Para cada consulta de debe escribir en el archivo:

- una línea que contenga tres $\#$ indicando el comienzo de una respuesta a cada consulta.
- para cada respuesta, una línea que contenga dos enteros i y j , separados por un espacio, que indican entre cuales punteros de salto se encontró el número a buscar. En caso que la consulta no tenga respuesta afirmativa se debe escribir un par de 0s.

Ejemplo

Dada la siguiente entrada:

```
86 35 75 13 51 23 37
45 84 17 50 39 56 60 114
64 72 94 77
80 47 90 59 110 62 -1 64 10 50 94 -2
```

Se debe generar la siguiente lista y sus punteros de salto:

```
13 17 23 35 37 39 45 47 50 51 56 59 60 62 64 72 75 77 80 84 86 90 94 110 114
*           *           *           *           *           *           *
```

y la salida es:

```
###
60 75
###
0 0
###
```

37 50

86 114

Restricciones

- La tarea debe ser codificada en Lenguaje C. No utilice funciones que no pertenecen al ANSI C.
- Para compilar utilice `gcc/mingw`.
- El desarrollo de la solución debe considerar el uso de las definiciones propuestas, listas enlazadas, funciones y paso de parámetros.
- Se debe respetar las especificaciones entregadas para la entrada y salida.
- La corrección incluye una interrogación.
- Solo se recibirán tareas fuera de plazo dentro de las 24 horas siguientes a la fecha de entrega. Nota máxima es un 5.0

Fecha de entrega : Jueves 22 de Noviembre, código fuente **tarea6.c** indicando en su interior como comentario su nombre y rut, enviar por email hasta las 24h00.
email grupo 1 <ici142-1@inf.ucv.cl>
email grupo 2 <ici142-2@inf.ucv.cl>