

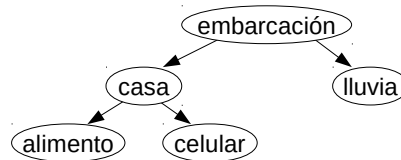
## CC3301 Programación de Software de Sistemas

### Tarea 2 – Semestre Primavera 2017 – Prof.: Luis Mateu

Un diccionario se implementa mediante un árbol de búsqueda binaria (ABB) almacenado en el archivo *dicc.txt* en el siguiente formato:

1	2	embarcacion	todo tipo de artilugio capaz de navegar en el agua
3	4	casa	edificacion construida para ser habitada
-1	-1	lluvia	condensacion del vapor de agua contenida en las nubes
-1	-1	alimento	sustancia ingerida por un ser vivo
-1	-1	celular	aparato portatil de un sistema de telefonía celular

Cada línea del archivo es de 100 caracteres y corresponde a un nodo del ABB. Los primeros 10 caracteres corresponden al número de la línea en donde se encuentra el hijo izquierdo o -1 si es nulo. Los 10 siguientes caracteres son el número de línea del hijo derecho (o -1). Luego vienen 20 caracteres para la llave y 59 caracteres para el valor asociado. La línea se termina con un `\n` para completar los 100 caracteres. La primera línea es la número 0 y corresponde a la raíz del ABB. Siempre existe un espacio en blanco que separa una columna de la otra. El archivo del ejemplo representa el siguiente ABB:



Programa el comando *consultar* que despliega el valor asociado a una llave. Ejemplos de invocaciones y sus resultados son:

```
$ ./consultar dicc.txt casa
edificacion construida para ser habitada
$ ./consultar dicc.txt embarcacion
todo tipo de artilugio capaz de navegar en el agua
$ ./consultar dicc.txt nadaquever
palabra no encontrada
$
```

Ud. debe buscar *eficientemente* la llave en el archivo, lo que quiere decir que no puede buscar secuencialmente. Tenga cuidado en considerar que `strcmp("casa", "casa ")` es distinto de 0.

Para obtener el número almacenado en un string use la función `atoi`. Por ejemplo `atoi(" -123 ")` entrega -123 y `atoi(" 8 10")` entrega 8, es decir que `atoi` solo considera el primer número del string.

Ayuda: Use `fread` para buscar secuencialmente la palabra en el archivo, luego use `fseek` para reposicionar el archivo en la línea que debe reescribir o agregar y finalmente use `fwrite` para escribir en el archivo.

### Restricciones

- Ud. no puede cargar el archivo completo en memoria, porque podría no caber en ésta.
- Su tarea debe funcionar correctamente con el ejemplo de este enunciado.

### Recursos

Baje *t2.zip* de material docente en U-cursos y descomprímalo. El directorio *T2* contiene el *Makefile* que le servirá para compilar su tarea y el archivo *dicc.txt* para que haga las pruebas.

### Entrega

Entregue su tarea solo si compila sin arrojar warnings en la máquina *anakena.dcc.uchile.cl* y funciona correctamente con el ejemplo de este enunciado. Ud. debe entregar el archivo *consultar.c* por medio de U-cursos. Se descontará medio punto por día de atraso. No se consideran los días sábado, domingo o festivos.