

CC3301 Programación de Software de Sistemas – Tarea 3 – Semestre Primavera 2019 – Prof.: Luis Mateu

El archivo de texto *cola.txt* almacena una cola de prioridad en el formato que muestra la columna de la izquierda de la siguiente tabla:

| archivo <i>cola.txt</i> | Ejemplo de uso | <i>cola.txt</i> después |
|--|-------------------------------------|--|
| 4 pedro 8 juan 2 diego 7 ximena 4 ana 1 | \$./extraer cola.txt juan \$ | 3 pedro 8 ximena 4 diego 7 ximena 4 ana 1 |

La primera línea del archivo tiene 4 caracteres más el $\backslash n$ (cambio de línea) y corresponde a la cantidad de elementos almacenados en la cola. Cada una del resto de las líneas tiene 16 caracteres más el $\backslash n$. Los primeros 10 corresponden a un texto almacenado en la cola y los 6 restantes a su prioridad. En el ejemplo la mejor prioridad es 2. El archivo podría contener 0, 1 o más líneas sobrantes al final (de 16+1 caracteres) debido a elementos extraídos, pero solo al final. Su contenido es irrelevante.

Programe el comando *extraer* que recibe como único argumento el nombre del archivo que contiene la cola y extrae de ella el texto con la mejor prioridad, mostrándolo en pantalla. Para la extracción Ud. **debe** (i) leer secuencialmente todos los elementos de la cola, buscando el texto con mejor prioridad; (ii) mostrarlo en pantalla (recuerde que al mostrar un string con *printf*, ese string debe terminar con $\backslash 0$); (iii) usar *fseek* para actualizar el tamaño de la cola; y (iv) usar *fseek* para mover el último elemento de la cola a la línea que queda libre.

La columna del centro de la tabla de arriba muestra un ejemplo de uso del comando *extraer* y la columna de la derecha muestra el contenido de *cola.txt* después de la extracción. Observe que el i -ésimo elemento de la cola está en la posición $4+1+i*(16+1)$ en el archivo (partiendo con $i \equiv 0$). Está prohibido almacenar todo el contenido de la cola en la memoria porque podría ser demasiado grande.

Ayuda: Estas funciones le serán de utilidad.

```
int n= atoi("25 \n"); // n es 25
char s[6], d[6];
sprintf(s, "%-4d\n", n); // imprime en un string, no en un archivo,
// %-4d significa mostrar entero en 4 caracteres, alineado a la izquierda,
// s es el string "25 \n" de largo 5
memcpy(d, s, 4); // copia 4 bytes a partir de s en d
```

Recursos

Baje el archivo *t3.zip* y descomprímalo. Contiene el archivo *Makefile* que compila su tarea. Programe su solución en el archivo *extraer.c*. Además contiene el archivo *cola.txt* que almacena la cola de prioridad. Su tarea se probará con los siguientes comandos:

```
$ ./extraer cola.txt
juan
$ ./extraer cola.txt
ximena
$ ./extraer cola.txt
diego
$ ./extraer cola.txt
pedro
$ ./extraer cola.txt
error: cola.txt esta vacia
$ cat cola.txt
0
pedro      8
diego      7
diego      7
ximena     4
ana        1
$
```

Su tarea debe arrojar los mismos resultados. El archivo final **debe ser** el que muestra al final el comando *cat* de Unix. No deben aparecer caracteres extraños.

Entrega

Ud. debe entregar mediante U-cursos el archivo *extraer.c*. No incluya otros archivos por favor. No se aceptarán tareas que no funcionen exactamente como el ejemplo de más arriba en la máquina *anakena.dcc.uchile.cl*. Se descontará medio punto si la compilación de su tarea arroja algún *warning* en *anakena*. Se descontará medio punto por día de atraso (excepto sábados, domingos y festivos).