

En la realización de este trabajo práctico se ha trabajado de forma modularizada para su resolución, implementando distintos TDAs de forma separada de modo de dividir el gran problema que se presenta en pequeños subproblemas. Establecimos entonces distintos TDAs a utilizar.

## Manual del Programador

Entre los principales cuya explicación merece un lugar en este “Manual de Programador” se encuentran:

1. **SYSTEM**
2. **ANTENNA**
3. **CELLPHONE**
4. **CALL**
5. **MESSAGE**
6. **INDEX**

Se han utilizado además otros TDA: LoadFile para cargar el archivo, cancelledCallsPerAntena para modularizar aun mas el problema y el conocido TDA Lista que se utiliza en la mayoría de los TDA mencionados anteriormente, pero cuya explicación se obvió.

Como este es un manual para el programador, explicamos a continuación la forma en que implementamos los TDAs mencionados:

### 1. SYSTEM

—> Un **sistema** se encarga de establecer las relaciones correspondientes entre las antenas del sistema, los celulares del sistema (que a su vez están conectados a distintas antenas) y las llamadas en progreso (asociadas a distintos celulares conectados a antenas).

Para ello, cuenta con una *lista de antenas* (ver **2. ANTENNA**) , una *lista de celulares* (ver **3.CELLPHONE**), una *lista de las llamadas* en progreso (ver **4. CALL**) y variables que apuntan a el celular que:

- \*) *hablo más*
- \*) *llamó más*
- \*) *le hablaron más*
- \*) *llamaron más*
- \*) *más recibió ocupado*
- \*) *más dio ocupado*

Un sistema se construye inicializando la lista de antenas, la lista de celulares y la lista de llamadas en progreso, reservándoles memoria dinámica. Todas las variables se inicializan apuntando a NULL.

Así mismo cuenta con métodos para:

- \*) Agregar una antena a la lista de antenas del sistema.
  - \*) Agregar un celular a la lista de celulares del sistema. En caso de que las variables mencionadas todavía no apunten a ningún celular, van a apuntar a este.
  - \*) Iniciar una llamada entre dos celulares que se encuentran en el sistema indicando el minuto en el que se inicia. Se crea un objeto de tipo "Call" (ver **4.CALL**) reservándole memoria dinámica.
- Se aprecian los casos en que:
- Los dos celulares están conectados

- > Se agregan las antenas a las que están conectadas los celulares a la lista de antenas involucradas de la llamada.
  - > Se cambia el estado de la llamada a EN PROGRESO y el estado de los celulares a HABLANDO
  - > Se incrementa el numero de llamadas realizadas para el celular iniciador de la llamada y el numero de llamadas recibidas para el celular receptor de la llamada
  - El celular receptor no puede atender la llamada (su estado es HABLANDO)
    - > Se incrementa el numero de llamadas realizadas rechazadas para el celular iniciador de la llamada y el numero de llamadas recibidas rechazadas para el celular receptor de la llamada
    - > Se realiza un chequeo de los celulares que mas recibieron ocupado y de los que mas estuvieron ocupado para ver si se tiene que cambiar.
  - El celular iniciador esta esperando a ser conectado
    - > Se busca la antena en la cual esta conectado celular y se incrementa las llamadas canceladas por exceso de capacidad de esa antena
    - > Se cambia el estado de la llamada a terminado
- Se realiza un chequeo de los celulares que mas llamaron y de los que fueron mas llamados para ver si los tiene que cambiar.
- Se agrega esta llamada a la lista de llamadas en progreso.

\*) Terminar una llamada que se encuentra EN PROGRESO, indicando la llamada y el minuto de finalización de la misma.

Se busca el celular iniciador de la llamada en la lista de celulares, se cambia su estado de HABLANDO a CONECTADO y se aumenta el numero de llamadas salientes del mismo según la duración de la llamada. Lo mismo con el celular receptor y sus llamadas entrantes.

Se realiza un chequeo de los celulares que hablaron mas y de los que mas fueron hablados para ver si tienen que cambiarse.

Devuelve la duración de la llamada.

\*) Enviar todos los mensajes no enviados

Busca en la lista de celulares los celulares que tengan mensajes sin enviar, los remueve de su lista de mensajes sin enviar, busca en la lista de celulares al receptor y le agrega el mensaje a su lista de espera

\*) Encontrar llamada EN PROGRESO por celular iniciador

Busca la llamada en la lista de llamadas del sistema, devuelve un puntero a la llamada encontrada o NULL si no la encontró

\*) Conectar un celular a una antena, ambos pasados por parámetro

Si el celular ya esta conectado a una antena, lo desconecta antes de conectarlo a una nueva.

\*) Desconectar un celular de la antena a la que esta conectada

Se encuentra la antena a la que esta conectada el celular, se desconecta el mismo y se busca el primer celular de la lista de celulares en espera de esa antena, para cambiarle el estado de la llamada EN ESPERA a EN PROGRESO.

\*) Métodos para obtener punteros al celular que:

mas hablo, mas llamo, mas le hablaron, mas llamaron, mas ocupado recibio, mas ocupado estuvo

\*) Métodos para chequear que si los punteros recién mencionados deben seguir apuntando al mismo celular o cambiarse a otro.

\*) Métodos para obtener la lista de las antenas y la lista de los celulares del sistema

\*) Metodos para encontrar:

- un celular en la lista de celulares

- una antena en la lista de antenas
- la antena a la que esta conectada un celular
- una llamada en progreso en la lista de llamadas
- la llamada en progreso por celular iniciador en la lista de llamadas

## 2. ANTENNA

—> Una **antena** cuenta con una *lista de mensajes* (ver 5. MESSAGE), una *lista de celulares conectados* (ver 3. CELLPHONE), una *lista de celulares en espera* y variables que:

- apuntan al celular que:

- \*) *hablo mas*
- \*) *llamo mas*
- \*) *le hablaron mas*
- \*) *llamaron mas*
- \*) *mas recibió ocupado*
- \*) *mas dio ocupado*

- guardan información sobre:

- \*) el *máximo numero de conexiones* (capacidad de la antena)
- \*) el *identificador* de la antena
- \*) el *máximo numero de conexiones simultáneas*
- \*) el *numero de llamadas canceladas por exceso de capacidad*

Una antena se crea opcionalmente con un identificador y un máximo numero de conexiones pasados por parámetro. Otra opción es que se cree vacía (con las variables mencionadas recién en 0). Para ambos casos se crea inicializando las listas mencionado reservándoles memoria dinámica, con las variables correspondientes apuntando a NULL y el resto de las variables inicializadas en 0.

Así mismo cuenta con métodos para:

- \*) Métodos para obtener los números de:  
el identificador, el máximo numero de conexiones, la cantidad de conexiones activas que hay actualmente, el máximo numero de conexiones simultáneas, el numero de llamadas canceladas por exceso de capacidad
- \*) Métodos para obtener punteros al celular que:  
mas hablo, mas llamo, mas le hablaron, mas llamaron, mas recibio ocupado, mas dio ocupado
- \*) Métodos para obtener las listas de los celulares conectados a la antena y los celulares en espera
- \*) Conectar un celular a la antena. En caso de haber conexiones libres devuelve TRUE y agrega el celular a la lista de celulares conectados, cambiando el estado del mismo a CONECTADO y su ultima conexión a esta antena. En caso contrario devuelve FALSE y agrega al celular a la lista de celulares en espera.
- \*) Desconectar un celular de la antena. El celular se encuentra en la lista de celulares conectados (recorre la lista). Borra el celular de la lista y devuelve un puntero al celular desconectado.
- \*) Chequear que el celular este conectado. Devuelve TRUE si el celular esta conectado y FALSE si no lo esta. Para ver si esta conectado, recorre la lista de celulares para ver si el celular pasado por parámetro se encuentra en ella. Si lo encuentra deja al cursor de la lista apuntando al celular buscado, sino apunta a NULL.
- \*) Chequear si la antena esta llena o no
- \*) Guardar mensaje, indicando el numero del celular emisor (conectado a la antena), el numero del celular receptor, el cuerpo del mensaje y el minuto. Guarda un mensaje para posteriormente tener la opción de mandarlo a su remitente.
- \*) Incrementar el numero de llamadas por exceso de capacidad

### 3. CELLPHONE

—> Un **celular** cuenta con (ver **5. MESSAGE**) una *lista de mensajes en espera*, *lista de mensajes en bandeja de entrada*, *lista de mensajes en bandeja de salida*, *lista de mensajes no enviados*, *lista de nuevos mensajes*, (ver **4. CALL**) *lista de llamadas entrantes*, *lista de llamadas salientes* y variables que guardan información sobre:

- \*) El *numero del celular*
- \*) La *ultima conexión* (la ultima antena a la que estuvo conectada)
- \*) El *estado* del celular: CONECTADO, DESCONECTADO, ACTUALMENTE HABLANDO o ESPERANDO CONEXION
- \*) *Minutos de las llamadas salientes*
- \*) *Minutos de las llamadas entrantes*
- \*) *Numero de llamadas salientes rechazadas*
- \*) *Numero de llamadas entrantes rechazadas*
- \*) *Numero de llamadas salientes*
- \*) *Numero de llamadas entrantes*
- \*) *Archivo de ingreso*
- \*) *Archivo de egreso*

Un celular se construye opcionalmente con un numero que se asigna por parámetro (la otra opción es con el numero nulo) y creando los archivos de egreso e ingreso. Cuando se construye un celular se inicializan todas las listas mencionadas reservándoles memoria dinámica. El estado se inicializa en DESCONECTADO. Todas las demás variables numéricas se inicializan en 0.

Así mismo cuenta con métodos para:

- \*) Obtener el numero del celular, el estado del celular, la ultima conexión, los minutos de las llamadas salientes, los minutos de las llamadas entrantes, el numero de las llamadas salientes rechazadas, el numero de las llamadas entrantes rechazadas, el numero de las llamadas salientes y el numero de las llamadas entrantes.
- \*) Cambiar el estado del celular. Si el estado es cambiado a CONECTADO, entonces se habilita para que reciba nuevos mensajes
- \*) Recibir nuevos mensajes. El celular tiene que estar CONECTADO. Los mensajes en la lista de espera se transfieren a la lista de la bandeja de entrada.
- \*) Enviar mensajes indicando el receptor, el mensaje y el minuto en que se envía. Se crea un nuevo mensaje y se lo agrega a la lista de mensajes en bandeja de salida y a la lista de mensajes no enviados.
- \*) Agregar mensaje a la lista de espera. Cuando el celular este CONECTADO, los mensajes se agregaran a la lista de mensajes de entrada y nuevos mensajes.
- \*) Métodos para aumentar los minutos de las llamadas salientes y los minutos de las llamadas entrantes
- \*) Métodos para incrementar el numero de llamadas salientes rechazadas, entrantes rechazadas, salientes y entrantes
- \*) Cambiar la ultima conexión
- \*) Métodos para obtener las listas de mensajes en bandeja de salida, mensajes no enviados, mensajes en bandeja de entrada, mensajes en espera, nuevos mensajes, llamadas salientes, llamadas entrantes

## 4. CALL

—> Una **llamada** cuenta con una *lista de las antenas involucradas* en la llamada (ver **2.ANTENA**), y variables que guardan información sobre:

- \*) El *iniciador* de la llamada
- \*) El *receptor* de la llamada
- \*) El *minuto de inicio* de la llamada
- \*) El *minuto de finalización* de la llamada
- \*) El *estado* de la llamada: OCUPADA, EN PROGRESO, TERMINADA, EN ESPERA

Una llamada se construye con un iniciador, un receptor y un minuto de inicio pasados por parámetro. El minuto de fin queda establecido por default igual que el minuto de inicio y el estado de la llamada es OCUPADA. Al construirse se inicializa la lista de Antenas involucradas en la llamada, reservándole memoria dinamica.

Así mismo cuenta con métodos para:

- \*) *Finalizar la llamada* indicando el minuto de finalización. La llamada debe estar EN PROGRESO. Cambia el minuto de fin de llamada por el indicado y cambia el estado a TERMINADA.
- \*) Métodos para *obtener* la duración de la llamada, el numero del celular que inicio la llamada, el numero del celular que recibió la llamada, el estado de la llamada, el minuto de inicio de la llamada, el minuto de fin de la llamada y una lista de las antenas involucradas en la llamada
- \*) *Cambiar el estado de la llamada*
- \*) *Agregar el ID de una antena a la lista de antenas involucradas*

## 5. MESSAGE

—> Un **mensaje** cuenta con un *cuerpo*, el *numero del receptor*, el *numero del emisor* y el *minuto* en que se envía el mensaje.

Un mensaje se crea opcionalmente con un cuerpo, un numero de emisor, un numero de receptor y un minuto asignados por parámetro. La otra opción es con un cuerpo vacío y un numero de emisor, numero de receptor y minutos inicializados en 0.

Así mismo cuenta con métodos para *obtener* el cuerpo del mensaje, el numero del receptor, el numero del emisor y el minuto en el que se envía el mensaje.

## 6. INDEX

Como este TDA no es de gran importancia en lo que es el funcionamiento del sistema de telefonía celular, explicamos su propósito a continuación pero sin entrar en detalles.

Un index es un indice cuya función principal es la interfaz con el usuario. Posee los métodos y variables necesarias para presentar en pantalla los menús con las opciones correspondientes. Según lo que escoja el usuario, el index provee las instrucciones necesarias para imprimir en pantalla los ítems solicitados por el usuario