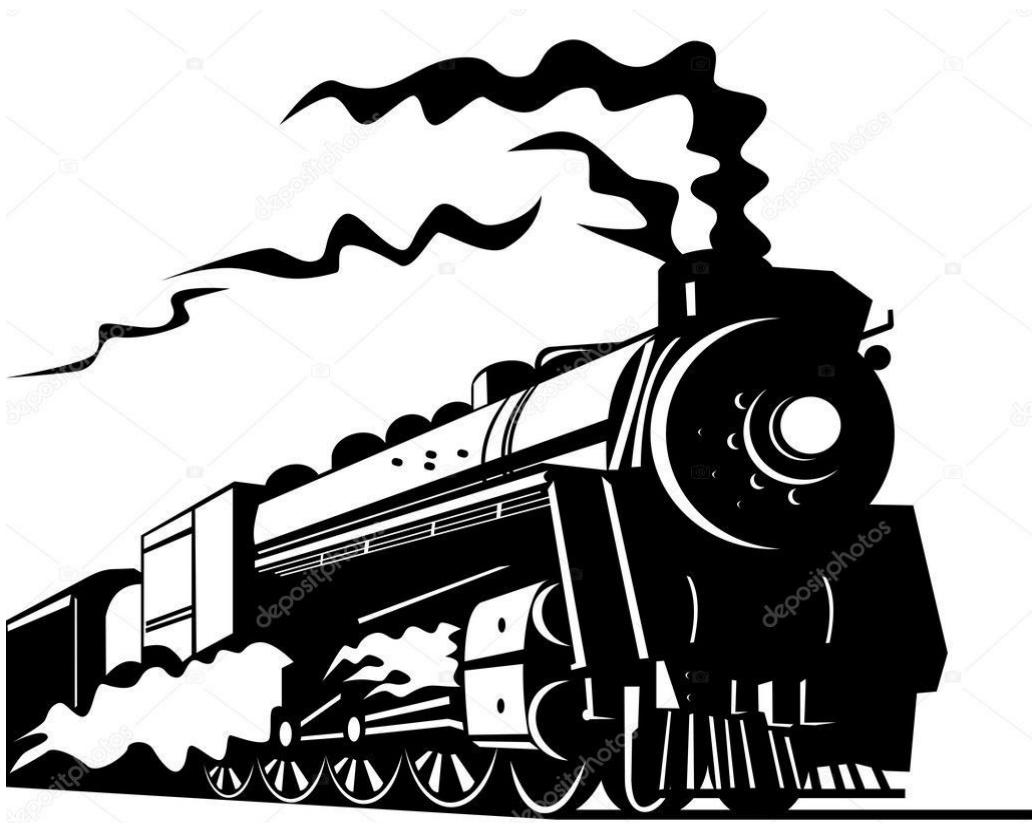


Sistemas Operativos

Trabajo Práctico 2019

Trenes



Profesor: Ing. Mostovoi Alejandro

Integrantes del Grupo:

Fernandez Gonzalo

Palacios Facundo

Pignatta Mauro

Pirrota Ezequiel

Ugobono Alejandro

Wiñar Mariano

Índice

Estructura de Datos	3
Interfaz gráfica	3
Variable global para la interfaz grafica	3
Estructura del tipo enum para los colores a utilizar	3
Funciones y Aplicaciones	3
void initUserInterface(ST_APP_WINDOW *)	3
void drawUserInterface(ST_APP_WINDOW *)	3
void printWindowTitle(WINDOW *pWin, const char * message)	4
void printLog(ST_APP_WINDOW *pWindow, const char * message, COLOUR	4
void printRegistro(ST_APP_WINDOW *pWindow, const char *message,	
COLOUR colour)	4
int printEstadoTrenes(ST_APP_WINDOW *pWin , TREN trenes[])	4
void unInitUserInterface(ST_APP_WINDOW *)	4
void printEstadoEstaciones(ST_APP_WINDOW *pWin, ESTACION est[])	4
void printHelp(ST_APP_WINDOW *pAppWin)	5
void clearWindow(WINDOW *pWin)	5
void clearCmdWindow(WINDOW *pWin)	5
void InterfazGrafica()	5

Interfaz de Estación

Es la conexión funcional entre los dos procesos, en este caso los Trenes como Cliente y las Estaciones como Servidores de esos Clientes, es decir una relación servidor/cliente, proporcionándonos así la comunicación entre ambos, el envío de datos e información necesario para el funcionamiento total del programa, además de las funciones relacionadas al dibujo en pantalla a través de la interfaz gráfica con ncurses

1- Estructuras de Datos:

1.1- Interfaz gráfica

```
typedef struct {  
    WINDOW *pAppFrame;  
    WINDOW *pLogFrame;  
    WINDOW *pLogWindow;  
    WINDOW *pCmdFrame;  
    WINDOW *pCmdWindow;  
    WINDOW *pRegFrame;  
    WINDOW *pRegWindow;  
} ST_APP_WINDOW;
```

1.2- Variable global para la interfaz grafica

```
ST_APP_WINDOW pWin;
```

1.3- Estructura del tipo enum para los colores a utilizar

```
typedef enum {RED=1, GREEN, BLUE, WHITE, YELLOW, CYAN,} COLOUR;
```

2- Funciones y Aplicaciones:

2.1- void initUserInterface(ST_APP_WINDOW *):

Aplicación:

Crea las ventanas de la app. Asocia colores con las ventanas, Crea en pantalla un grupo de ventanas con la siguiente estructura:

Log: muestra mensajes,

Cmd: permite el ingreso de comandos al usuario.

Parámetros:

ST_APP_WINDOW * puntero a estructura que contiene las ventanas.

2.2- void drawUserInterface(ST_APP_WINDOW *):

Aplicación:

Dibuja el Marco de la App, Ventana Log, Ventana Cmd, Ventana Reg,

Parámetros:

ST_APP_WINDOW * puntero a estructura que contiene las ventanas.

2.3- void printWindowTitle(WINDOW *pWin, const char * message):

Aplicación:

Toma el Título a imprimir,

Parámetros:

- * pWin Variable global para la interfaz grafica,
- * message es la Estación y su numero como titulo.

2.4- void printLog(ST_APP_WINDOW *pWindow, const char * message, COLOUR colour):

Aplicación:

Imprime en la pantalla tipo Log en la parte superior,

Parámetros:

- * pWindow puntero a la estructura que contiene las ventanas,
- * message contiene el mensaje a imprimir,
- colour de la estructura tipo enum con los colores a utilizar.

2.5- void printRegistro(ST_APP_WINDOW *pWindow, const char *message, COLOUR colour):

Aplicación:

Imprime en la parte derecha de la ventana si se conecto el tren, se utiliza en funcEstaciones.c,

Parámetros:

- * pWindow puntero a la estructura que contiene las ventanas,
- * message contiene el mensaje a imprimir,
- COLOUR colour es de la estructura tipo enum con los colores a utilizar.

2.6- int printEstadoTrenes(ST_APP_WINDOW *pWin , TREN trenes[]):

Aplicación:

Imprime el Estado completo del Tren con todas sus variables,

Parámetros:

- * pWin Variable global para la interfaz grafica,
- TREN trenes es el vector de trenes,

Retorna:

Un entero.

2.7- void unInitUserInterface(ST_APP_WINDOW *):

Aplicación:

Inicia el usuario de la interfaz grafica,

Parámetros:

- * variable global para la interfaz grafica.

2.8- void printEstadoEstaciones(ST_APP_WINDOW *pWin, ESTACION est[]):

Aplicación:

Imprime el Estado completo de la Estación,

Parámetros:

- * pWin Variable global para la interfaz grafica,
- ESTACION est es el vector de estaciones.

2.9- void printHelp(ST_APP_WINDOW *pAppWin):

Aplicación:

Imprime la Ayuda del programa,

Parámetros:

* pAppWin representa la Variable global para la interfaz grafica.

2.10- void clearWindow(WINDOW *pWin):

Aplicación:

Limpia la ventana de comandos y hace un refresco de pantalla,

Parámetros:

* pWin variable global, ventana de comandos,

Retorna:

ERR_OK.

2.11- void clearCmdWindow(WINDOW *pWin):

Aplicación:

Limpia la ventana de comandos,

Parámetros:

* pWin variable global, ventana de comandos,

Retorna:

ERR_OK.

2.12- void InterfazGrafica():

Aplicación:

Para el hilo que se encarga de la interfaz grafica.