Universidad de San Carlos de Guatemala Facultad de Ingeniería Sistemas Operativos 2

PROYECTO

SCRIPT PARA MONTAR EL SISTEMA DE ARCHIVOS

Comandos para compilar el archivo de C

gcc -Wall `pkg-config fuse --cflags` fuse201504429Proyecto.c -o fuse201504429Proyecto `pkg-config fuse –libs`

donde fuse201504429Proyecto.c es el archivo de C fuse201504429Proyecto es el archivo de salida

Comandos para Montar el Sistema de Archivos

./fuse201504429Proyecto -d -f -s /montar

donde ./fuse201504429Proyecto es el compilado de C /montar es el directorio donde se montar

> sudo mkdir /filesystem_201504429 sudo chmod +777 /filesystem_201504429 Crear una carpeta inicial

sudo mkdir /montar
sudo chmod +777 /montar
Crear una carpeta para montar el sistema de archivos

Comandos para Desmontar el Sistema de Archivos

fusermount -u /montar

Desmonta el sistema de archivos

FUNCIONES

FUNCION PARA INTERRUMPIR LA ELIMINACION DE ARCHIVOS

```
static int xmp_unlink(const char *path)
{
      char cadena[strlen(path)+1];
      strcpy(cadena, path);
      char *ptrToken; // crea un apuntador char
      char nombrearch[strlen(path)+1];
      ptrToken = strtok( cadena, "/" );
      //int res;
      while ( ptrToken != NULL ) {
      strcpy(nombrearch, ptrToken);
      ptrToken = strtok( NULL, "/" );
    }
}
```

ESTA PRIMERA PARTE COPIA EL PATH A OTRA VARIABLE Y LA RECORRE CON UN STRTOK

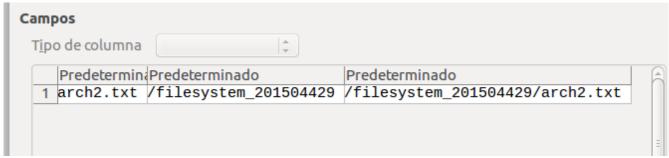
ESTA PARTE ACUMULA UN PUNTERO CON LOS FOLDERS DESDE EL / HASTA EL NOMBRE DEL ARCHIVO

LA FUNCION DE RENAME SE ENCARGA DE MOVER EL ARCHIVO A LA PAPELERA DE RECICLAJE

```
int tamNuevaL = (strlen(nombrearch))+1+(strlen(acumulado))+1+(strlen(path))+1;
char nuevalinea[tamNuevaL];
strcpy(nuevalinea, nombrearch);
strcat(nuevalinea, ",");
strcat(nuevalinea, acumulado);
strcat(nuevalinea, ",");
strcat(nuevalinea, path);
strcat(nuevalinea, "\n");
nuevalinea[tamNuevaL] = '\0';
int fd = open(archivo_log, O_WRONLY | O_APPEND);
if (fd == -1){
              fprintf(stderr, "Error Abriendo Archivo LOG!\n");
              return -errno;
write(fd, nuevalinea, tamNuevaL);
close(fd);
tamanoBufferLog = tamanoBufferLog + tamNuevaL;
int fdh = open(archivo_historial, O_WRONLY | O_APPEND);
if (fdh == -1){
              fprintf(stderr, "Error Abriendo Archivo Historial!\n");
              return -errno:
char lineahis[tamNuevaL + sizeof(",ELIMINAR")];
strcpy(lineahis, nombrearch);
strcat(lineahis, ",");
strcat(lineahis, acumulado);
strcat(lineahis, ",");
strcat(lineahis, path);
strcat(lineahis, ",ELIMINAR");
strcat(lineahis, "\n");
```

```
lineahis[tamNuevaL + sizeof(",ELIMINAR")] = '\0';
write(fdh, lineahis, sizeof(lineahis) - 1);
close(fdh);
return 0;
}
```

ESTA ULTIMA PARTE ABRE EL ARCHIVO DE SALIDA Y AGREGA UNA ENTRADA CON EL ARCHIVO ELIMINADO Y SU PATH CORRESPONDIENTE



CODIGO PARA ITERAR EL ARCHIVO LOG

```
<u>ις της restaura</u>rοπολοποζοπαι *τίπεα, της τιροχ
char *nombrearch;
char *pathPapeleraArchivo;
char *pathAntiguo;
char *ptrTokenInterno;
char *lecturaLinea = (char *)calloc(strlen(linea), sizeof(char));
strcpy(lecturaLinea, linea);
ptrTokenInterno = strtok(lecturaLinea, ",");
int conteo = 1;
int restaurar = 0;
char *bufNuevoCSV = (char *)calloc(tamanoBufferLog, sizeof(char));
strcpy(bufNuevoCSV, "");
while ( ptrTokenInterno != NULL ) {
        if(1 == conteo){
            nombrearch = (char *)calloc(strlen(ptrTokenInterno)+1, sizeof(char));
            strcpy(nombrearch, ptrTokenInterno);
            pathPapeleraArchivo = (char *)calloc(strlen(pathRecycle)+1+strlen(ptrToke
            strcpy(pathPapeleraArchivo, pathRecycle);
            strcat(pathPapeleraArchivo, "/");
            strcat(pathPapeleraArchivo, ptrTokenInterno);
```

USANDO EL SISTEMA DE ARCHIVOS

Crear y eliminar archivo

```
el-PC:/montar/filesystem_201504429$ echo contenido>arch2.txt
el-PC:/montar/filesystem_201504429$ rm arch2.txt
el-PC:/montar/filesystem_201504429$
```

Contenido de la Papelera de reciclaje

```
miguel@miguel-PC:/montar/filesystem_201504429/recycle$ ls
arch2.txt
miguel@miguel-PC:/montar/filesystem_201504429/recycle$
```

Comando para restaurar una carpeta

truncate i --size 1

Comando para restaurar TODO

truncate i --size 2