

## ARCHIVOS Y DIRECTORIOS EN GNU/LINUX.

### Manejo de archivos en GNU/Linux

Como se explicó en las clases teóricas, el manejo de archivos se lleva a cabo mediante llamadas POSIX de manejo de archivos.

Estas funciones (en FreePascal) se encuentran en la unit OldLinux:

<b>fdOpen</b>	Abre/crea un descriptor de archivo.
<b>fdRead</b>	Lee datos desde un descriptor de archivo.
<b>fdWrite</b>	Escribe en un descriptor de archivo.
<b>fdSeek</b>	Mueve el puntero de posición de un archivo.
<b>fdClose</b>	Cierra un descriptor de archivo.

Para ver otras funciones de manejo de archivos vea la documentación de FreePascal.

Ingrese al sistema como **usuario no privilegiado**. Cree un directorio donde almacenar los programas de este práctico dentro de su directorio home. Luego sitúese en el directorio recién creado.

Escriba el siguiente programa, nómbrelo "**prog\_0801.pp**" y compílelo.

```
Program prog_0801;

uses OldLinux;

Var fd : Longint;
    datos: String;

begin
  fd := FDOpen(ParamStr(1),Open_WrOnly OR Open_Creat);
  if fd > 0 then
    Writeln('Escriba lo que desea en el archivo:');
    Readln(datos);
    begin
      if (Length(datos)+1) <> (FDWrite(fd,datos,Length(datos)+1)) then
        Writeln('Error al escribir en el archivo!!!');
      FDClose(fd);
    end;
  end.
```

Cree un archivo vacío. Ejecute el programa pasándole como parámetro el archivo recién creado e ingrese algo de texto (al menos 20 caracteres) para escribir dentro del archivo. Verifique que la información se grabó en el archivo.

(1) Ejecute el programa nuevamente pasándole como parámetro un archivo inexistente. Se ejecutó igual? Grabó la información dentro del archivo? Por qué?

(2) Ejecute el programa nuevamente pasándole como parámetro el primer archivo e ingrese otra información (al menos 20 caracteres) para grabar en el archivo. Verifique que la información se grabó en el archivo. Qué ocurrió con la información que habíamos grabado en la primera ocasión? Por qué?

La documentación de FreePascal (rtl.pdf) pueden ser de gran ayuda para responder a las preguntas anteriores.

Escriba el siguiente programa y nómbrelo “**prog\_0802.pp**” y compílelo.

```
Program prog_0802;

uses OldLinux;

var fd : Longint;
    datos: String;

begin
    fd := fdOpen(ParamStr(1),Open_RdOnly);
    if fd>0 then
        begin
            if fdRead(fd,datos,10) < 0 then
                begin
                    Writeln ('Error leyendo archivo!!!');
                    Halt(2);
                end;
            Writeln(datos);
        end;
    fdClose(FD);
end.
```

(3) Ejecute el programa pasándole como parámetro el archivo de texto del ejercicio anterior. Observe los resultados. Por qué no mostró la totalidad del archivo?

### Manejo de archivos y directorios en GNU/Linux

En un nivel más bajo se puede acceder mediante funciones POSIX, a operaciones generales de manejo de archivos, directorios y sistemas de archivos.

Estas funciones (en FreePascal) se encuentran en la unit OldLinux:

<b>Access</b>	Chequea los permisos de acceso a un archivo
<b>Chown</b>	Cambia el dueño de un archivo.
<b>Chmod</b>	Cambia los permisos de acceso de un archivo.
<b>FLock</b>	Bloquea un archivo.
<b>FStat</b>	Brinda información sobre un archivo.
<b>FRename</b>	Renombra un archivo.
<b>LStat</b>	Brinda información sobre un enlace (link).
<b>Link</b>	Crea un enlace (link).
<b>ReadLink</b>	Lee el contenido de un enlace simbólico.
<b>SymLink</b>	Crea un enlace simbólico.
<b>UnLink</b>	Elimina un archivo.
<b>Utime</b>	Cambia la fecha de un archivo.
<b>OpenDir</b>	Abre un directorio para lectura.
<b>CloseDir</b>	Cierra un directorio,
<b>ReadDir</b>	Lee una entrada de directorio.
<b>TellDir</b>	Indica la entrada de directorio actual.
<b>SeekDir</b>	Se ubica en una entrada determinada del directorio.
<b>FSSStat</b>	Brinda información sobre el filesystem.

Para ver otras funciones de manejo de archivos, directorios y sistemas de archivos vea la documentación de Freepascal.

(4) Utilizando la documentación de Freepascal emplee la función **Access** para reescribir los programas anteriores de manera que verifiquen que el archivo existe y que el usuario tiene los privilegios para realizar las operaciones sobre el mismo.

Escriba el siguiente programa y nómbrelo “**prog\_0803.pp**” y compílelo.

```
program prog_0803;
uses OldLinux;
var info : Stat;
begin
  if not FStat(ParamStr(1),info) then begin
    Writeln('Fallo la llamada a FStat!!!');
    halt (1);
  end;
  writeln ('Resultados del fstat del archivo:');
  writeln ('I-nodo           : ',info.ino);
  writeln ('Modo (Tipo y Permisos) : ',info.mode);
  writeln ('Numero de links       : ',info.nlink);
  writeln ('User ID del dueño     : ',info.uid);
  writeln ('Group ID del dueño    : ',info.gid);
  writeln ('Tipo de dispositivo inodo : ',info.rdev);
  writeln ('Tamaño del Bloque      : ',info.blksize);
  writeln ('Tamaño en Bytes        : ',info.size);
  writeln ('Nro. de Bloques        : ',info.blocks);
  writeln ('Ultimo acceso          : ',info.atime);
  writeln ('Ultima modificacion    : ',info.mtime);
  writeln ('Ultimo cambio          : ',info.ctime);
end.
```

(5) Pruebe el programa pasándole como parámetro distintos archivos. Dónde está almacenada esta información de cada archivo?  
Puede obtener más información sobre el registro Stat mediante la pagina man del mismo Stat(2).

Escriba el siguiente programa y nómbrelo “**prog\_0804.pp**” y compílelo.

```
program prog_0803;
uses OldLinux;
var info : Stat;
begin
  if not LStat(ParamStr(1),info) then begin
    Writeln('Fallo la llamada a LStat!!!');
    halt (1);
  end;
  writeln ('Resultados del LStat del archivo:');
  writeln ('I-nodo           : ',info.ino);
  writeln ('Modo (Tipo y Permisos) : ',info.mode);
  writeln ('Numero de links       : ',info.nlink);
  writeln ('User ID del dueño     : ',info.uid);
  writeln ('Group ID del dueño    : ',info.gid);
  writeln ('Tipo de dispositivo inodo : ',info.rdev);
  writeln ('Tamaño del Bloque      : ',info.blksize);
  writeln ('Tamaño en Bytes        : ',info.size);
  writeln ('Nro. de Bloques        : ',info.blocks);
  writeln ('Ultimo acceso          : ',info.atime);
  writeln ('Ultima modificacion    : ',info.mtime);
  writeln ('Ultimo cambio          : ',info.ctime);
end.
```

Cree un archivo de texto llamado **prueba.txt** y agreguele algo de contenido.  
Ejecute la siguiente línea:

```
ln -s prueba.txt prueba1.txt
```

Luego ejecute:

```
./prog_0803 prueba.txt
./prog_0803 prueba1.txt
./prog_0804 prueba1.txt
```

(6) A qué se deben las diferencias y similitudes en las tres salidas? (preste especial atención a los campos i-nodo y tamaño)

Escriba el siguiente programa y nómbrelo **"prog\_0805.pp"** y compílelo.

```
program prog_0805;

uses OldLinux;

var archivo: Stat;

begin
  if FStat(ParamStr(1),archivo) then
  begin
    if S_ISLNK(archivo.mode) then
      Writeln ('El archivo es un enlace...');
    if S_ISREG(archivo.mode) then
      Writeln ('El archivo es un archivo regular...');
    if S_ISDIR(archivo.mode) then
      Writeln ('El archivo es un directorio...');
    if S_ISCHR(archivo.mode) then
      Writeln ('El archivo es un dispositivo de caract...');
    if S_ISBLK(archivo.mode) then
      Writeln ('El archivo es un dispositivo de bloques...');
    if S_ISFIFO(archivo.mode) then
      Writeln ('El archivo es una canieria (Pipe)...');
    if S_ISSOCK(archivo.mode) then
      Writeln ('El archivo es un socket...');
  end;
end.
```

(7) Pruebe el programa pasándole como parámetro archivos de distintos tipos (enumerelos y diga cuál fue el resultado). Dónde está almacenada esta información de cada archivo?

Escriba el siguiente programa y nómbrelo **"prog\_0806.pp"** y compílelo.

```
program prog_0806;

uses OldLinux;

var directorio : PDir;
    entrada : PDirent;

begin
  directorio := OpenDir(ParamStr(1));
  repeat
    entrada := ReadDir(directorio);
    if entrada <> nil then
    begin
      Writeln ('-----');
      Writeln ('Entrada No      : ',TellDir(directorio));
      Writeln ('Nombre          : ',pchar(@entrada^.name[0]));
      Writeln ('I-nodo           : ',entrada^.ino);
      Writeln ('Offset          : ',entrada^.off);
      Writeln ('Long. Nombre    : ',entrada^.reclen);
    end;
  until entrada = nil;

  CloseDir (directorio);
end.
```

Pruebe el programa pasándole como parámetro distintos directorios.

Escriba el siguiente programa y nómbrelo “**prog\_0807.pp**” y compílelo.

```
program prog_0807;
uses OldLinux;
var ubicacion : String;
    fsinfo : StatFS;
begin
  ubicacion:= '.';
  if not FSStat(ubicacion,fsinfo) then
    begin
      Writeln('Fallo el FSStat. Error No : ',LinuxError);
      Halt(1);
    end;
  Writeln ('Tipo de FS           : ',fsinfo.fstype);
  Writeln ('Tamaño de bloque    : ',fsinfo.bsize);
  Writeln ('Bloques libres          : ',fsinfo.bfree);
  Writeln ('Bloques disponibles      : ',fsinfo.bavail);
  Writeln ('Archivos                  : ',fsinfo.files);
  Writeln ('Descriptores libres       : ',fsinfo.ffree);
  Writeln ('Identificacion de FS     : ',fsinfo.fsid);
  Writeln ('Long. del Nombre         : ',fsinfo.namelen);
end.
```

(8) Ejecute el programa. Ubíquese en el directorio raíz (/) y vuelva a ejecutar el programa. Ahora sitúese en el directorio **/proc** y ejecute nuevamente el programa. Sobre qué filesystems nos está mostrando información en cada ocasión? Explique el significado de cada resultado.

Puede obtener más información sobre el registro StatFS mediante la pagina man del mismo StatFS(2).

(9) Modifique el programa anterior (nómbrelo “**prog\_0808.pp**”) para que muestre el espacio total, ocupado y libre (en bytes) en el sistema de archivos?

(10) Realice su propia versión del comando **ls** empleando las llamadas POSIX de gestión de archivos y directorios. Nómbrelo “**prog\_0809.pp**”, compílelo y pruebelo listando distintos directorios. Para mantener la sencillez, el comando deberá cumplir con las siguientes características:

- Deberá mostrar (en este orden): permisos del usuario actual (**rwX**), tamaño en Bytes y nombre de cada archivo del directorio en cuestión. (Uno por línea)
- Deberá además, permitir diferenciar archivos de ciertos tipos: ejecutables (por el usuario actual), enlaces simbólicos, directorios y dispositivos.
- Deberá mostrar al final el número de archivos listados.
- Deberá mostrar mensajes significativos en caso de no poder realizar el listado.

Confeccione un informe **original, conciso y completo** donde se dé respuesta a las preguntas y consignas precedidas por un número encerrado entre paréntesis. Éste informe deberá ser confeccionado y entregado por cada grupo que llevó a cabo las actividades. La fecha límite de entrega es el **14/10/2008**. La longitud máxima del informe es de dos páginas (sin contar las líneas correspondientes a código fuente).