

ARCHIVOS Y SISTEMAS DE ARCHIVOS.

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:

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

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

Ingresa 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_1001.pp"** y compílelo.

```
Program prog_1001;

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.
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 pueden ser de gran ayuda para responder a las preguntas anteriores.

Escriba el siguiente programa y nómbrelo "prog_1002.pp" y compílelo.

```
Program prog_1002;

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.
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:

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

FStat Brinda información sobre el filesystem.

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

Escriba el siguiente programa y nómbrelo "prog_1003.pp" y compílelo.

```
program prog_1003;

uses OldLinux;

var info : Stat;

begin
  if not FStat(ParamStr(1),info) then
    begin
      Writeln('Fallo la llamada a FStat!!!');
      halt (1);
    end;
  writeln;
  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 Bytes                 : ',info.size);
  writeln ('Tamaño de Bloque              : ',info.blksize);
  writeln ('Nro. de Bloques               : ',info.blocks);
  writeln ('Último acceso                 : ',info.atime);
  writeln ('Última modificación          : ',info.mtime);
  writeln ('Último cambio                 : ',info.ctime);

end.
```

(4) 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 página man del mismo Stat(2).

Escriba el siguiente programa y nómbrelo "prog_1004.pp" y compílelo.

```
program prog_1004;

uses OldLinux;

var archivo: Stat;

begin
  if LStat(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.
end.
```

(5) Pruebe el programa pasándole como parámetro archivos de distintos tipos. Dónde está almacenada esta información de cada archivo?

Escriba el siguiente programa y nómbrelo "prog_1005.pp" y compílelo.

```
program prog_1005;

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_1006.pp**” y compílelo.

```
program prog_1006;

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.
```

(6) 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 que filesystems nos está mostrando información en cada ocasión? Explique el significado de cada resultado. Cuál es el espacio libre en cada sistema de archivos?

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

(7) Realice su propia versión del comando **ls** empleando las llamadas POSIX de gestión de archivos y directorio. Nómbrelo “**prog_1007**”, compílelo y pruebelo listando distintos directorios. Para mantener la sencillez, el comando deberá cumplir con las siguientes características:

- Deberá mostrar (en este orden): i-nodo, tamaño en bytes y nombre de cada archivo del directorio en cuestión. (Uno por línea)
- Deberá enumerar al final el número de entradas del directorio.

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 **13/11/2007**. La longitud máxima del informe es de dos páginas (sin contar las líneas correspondientes a código fuente).