

Práctico 1: Archivos

Referencias

- Material de las clases Teóricas de Introducción a la Programación II.
- Bishop, Judy; Pascal Precisely, 3rd Edition, Addison-Wesley, 1993. Capítulo 8.

1) Dadas las siguientes versiones del programa SumarArchivo:

```
program SumarArchivo;
procedure abrirArch(var arch : File of Integer; nombre : String; var error : Boolean);
begin
    error := false;
    assign(arch, nombre);
    {$I-} { desactiva la verificación de errores de entrada/salida (en tiempo de ejecución) }
    reset(arch); { al no estar activada la detección de errores, se puede intentar abrir archivos inexistentes y no se recibirán mensajes de error }
    {$I+} { activa la verificación de errores entrada/salida }
    if ioresult <> 0 then {ioresult devuelve un 0 si la última operación de entrada/salida se realizó con éxito y <> 0 si hubo algún error}
        error := true;
end;
function sumarArch(var arch : File of Integer) : Integer;
var
    val, suma : Integer;
begin
    suma := 0;
    seek(arch, 0);
    while not eof(arch) do
        begin
            read(arch, val);
            suma := suma + val;
        end;
    sumarArch := suma;
end;
procedure escribirFinal(var arch : File of Integer; val : Integer);
begin
    seek(arch, fileSize(arch));
    write(arch, val);
end;
```

VERSIÓN 1

```
var
    arch : File of Integer;
    nombre : String;
    error : Boolean;
begin
    writeln('Ingresar nombre del archivo:');
    readln(nombre);
    abrirArch(arch, nombre, error);
    if not error then
        begin
            escribirFinal(arch, sumarArch(arch));
            close(arch);
        end
    else
        writeln('El archivo no existe');
    end.
```

VERSIÓN 2

```
var
    orig, sum : File of Integer;
    nombre : String;
    error : Boolean;
begin
    writeln('Ingresar nombre del archivo a sumar:');
    readln(nombre);
    abrirArch(orig, nombre, error);
    if not error then
        begin
            abrirArch(sum, 'sumas.dat', error);
            if error then
                rewrite(sum);
            escribirFinal(sum, sumarArch(orig));
            close(sum);
            close(orig);
        end
    else
        writeln('El archivo no existe');
    end.
```

a) Explique brevemente qué hace cada versión del programa.

b) Analice y compare el comportamiento de cada versión del programa al ejecutarla tres veces consecutivas. Suponga que el archivo utilizado, en todas las ejecuciones, es "origen.dat". El contenido inicial de origen.dat es: "4, 6, 10". El archivo "sumas.dat", al que hace referencia la versión 2, inicialmente no existe.

- 2) Teniendo en cuenta que los siguientes archivos son de números enteros, realice programas acorde a los siguientes incisos:
 - a) Dado un nombre de archivo, guardar en el archivo todos los números positivos ingresados por teclado.
 - b) Dado un nombre de archivo, mostrar por pantalla todos los elementos del mismo.
 - c) Dado un nombre de archivo de origen clonarlo con otro nombre; es decir, crear otro que tenga exactamente el mismo contenido pero un nombre distinto.
- 3) Teniendo en cuenta las suposiciones de los incisos a) y b) respectivamente, codifique una función que dado un número lo busque en un archivo de números y retorne la posición en la que lo encontró. Si no lo encuentra debe retornar -1.
 - a) Suponga que los elementos del archivo se encuentran desordenados.
 - b) Suponga que los elementos del archivo se encuentran ordenados. Tenga en cuenta que, si el archivo está ordenado, no es necesario recorrerlo completamente. Explique por qué.
- 4) Suponga que tiene un archivo donde cada registro es un arreglo de 5 enteros. Cada entero corresponde a la nota en la materia 1, 2, 3, 4 y 5 de los alumnos de primer año de una carrera. Cada registro del archivo corresponde a las notas de un alumno y el archivo contiene las notas de todos los alumnos de primer año de la carrera. Codifique un procedimiento (¿por qué no función?) que devuelva un arreglo con la nota promedio en cada una de las materias.

7	6	5	7	8
3	9	8	5	4
6	8	9	5	3
5	6	5	7	8
3	9	2	5	4
6	8	2	4	3

Archivo

5	8	5	6	5
---	---	---	---	---

Promedio de cada materia

- 5) Invierta el orden de los elementos de un archivo de caracteres.
- 6) Teniendo en cuenta que los siguientes archivos son de números enteros, realice procedimientos acorde a los siguientes incisos:
 - a) Recorra (una vez sola) un archivo de forma tal que si un número es mayor que el anterior los intercambie.
 - b) Utilizando el procedimiento anterior, ordene un archivo por el método de burbujeo sobre el mismo archivo.
- 7) Codifique un procedimiento para realizar una búsqueda binaria en un archivo (ordenado) de enteros. Compare la cantidad de accesos promedio de esta estrategia respecto a la del inciso 3.b.
- 8) En una universidad se tienen dos archivos con los números (enteros) de legajo de alumnos, uno posee los inscriptos en Análisis y el otro en Álgebra. Ambos archivos están ordenados por el n° de legajo. Confeccione un programa que genere otro archivo con los n° de legajo de los alumnos que están inscriptos en ambas materias.