



## Resolución de Problemas y Algoritmos

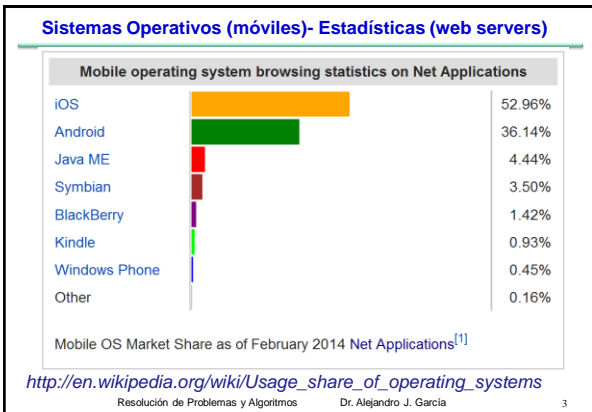
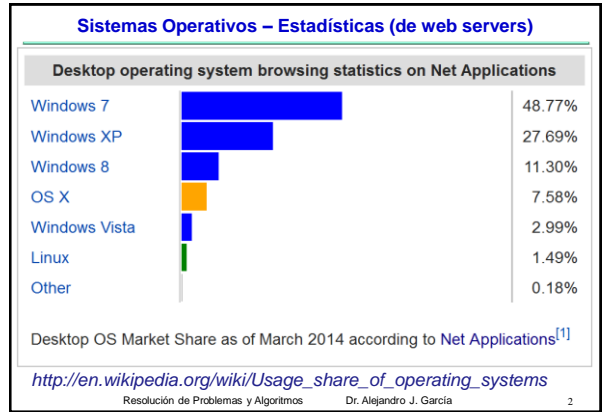
### Clase 8: Sistemas Operativos. Almacenamiento en Memoria. Archivos secuenciales en Pascal.



**Dr. Alejandro J. García**  
http://cs.uns.edu.ar/~ajg



Departamento de Ciencias e Ingeniería de la Computación  
Universidad Nacional del Sur  
Bahía Blanca - Argentina



### Sistemas Operativos - Estadísticas

Source	Date	Microsoft Windows					Apple		Linux kernel		
		8	7	Vista	XP	WP&RT	Other	OS X	iOS	Linux	Android
Net Market Share <sup>[6][7][8]</sup>	Mar-14	▲8.43%	▲36.37%	▼2.23%	▼20.65%	▲0.18%	0.19%	▲5.65%	▲13.55%	▲1.11%	▲10.53%
StatCounter Global Stats <sup>[9]</sup>	Mar-14	▲8.66%	▼38.35%	▼2.58%	▼13.06%	▲0.63%	▼0.14%	▲6.02%	▲9.88%	—	▲12.94%
W3Counter <sup>[10]</sup>	Mar-14	▲8.34%	▼39.04%	▼2.63%	▼9.30%	—	—	▼9.30%	▲9.82%	▲2.00%	▲7.86%
AT Internet <sup>[11]</sup>	Feb-14	▲8.00%	▲43.30%	▼4.70%	▼14.40%	—	—	▼6.60%	▲12.70%	▲1.00%	▲8.20%
Wikimedia <sup>[12]</sup>	Mar-14	▲6.29%	▲27.73%	▼2.03%	▼7.82%	▼0.67%	0.10%	▼5.78%	▲32.52%	▲1.37%	▲11.25%

[http://en.wikipedia.org/wiki/Usage\\_share\\_of\\_operating\\_systems](http://en.wikipedia.org/wiki/Usage_share_of_operating_systems)

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 5

- ### Sistemas operativos
- Es importante el "concepto" y no el "producto" porque en su carrera y trabajo profesional usará muchos sistemas operativos.
  - Estos son algunos de los que he usado hasta ahora:
    - RSTS (en PDP-11) y VMS (en VAX)
    - MS-DOS, Windows (3, NT, XP, Vista, 7, 8),
    - Unix, Linux, AIX, Symbian, Android,
  - Puede mirar más sobre sistemas operativos en:
    - [http://es.wikipedia.org/wiki/Sistema\\_operativo](http://es.wikipedia.org/wiki/Sistema_operativo)
    - [http://en.wikipedia.org/wiki/Mobile\\_operating\\_system](http://en.wikipedia.org/wiki/Mobile_operating_system)
- ¿Qué es un Sistema Operativo?
- Resolución de Problemas y Algoritmos Dr. Alejandro J. García 6

El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:  
 "Resolución de Problemas y Algoritmos. Notas de Clase". Alejandro J. García. Universidad Nacional del Sur. (c) 2014

### Conceptos: Sistema Operativo (Operating System)

- Un **sistema operativo (SO)** es un programa que gestiona los recursos de hardware y provee servicios a los programas de aplicación. Se ejecuta en modo privilegiado respecto de los restantes programas.
- Es un programa que actúa como un intermediario entre un usuario y el hardware de la computadora.

Cada carrera tiene una materia para SO:

- Ing: "Sistemas operativos"
- Lic: "Sistemas operativos y distribuidos"

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 7

### Conceptos: Sistema Operativo (SO)

- Por ejemplo, un sistema operativo conoce los detalles del hardware del almacenamiento secundario (como un disco rígido) y provee servicios a los programas de aplicación para el **manejo de archivos**.
- Puede haber archivos de video, de música, de imágenes, de texto, de programas ejecutables, etc.
- Observe que el nombre del archivo le permite al sistema operativo asociarlo con una aplicación.

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 8

### Nombres de archivos en un SO

- Un nombre válido para un archivo dependerá del SO.
- Por ejemplo, en las primeras versiones del sistema operativo MS-DOS (1981-1995), un nombre de archivo tenía el formato: **nnnnnnn.EEE** (i.e., 8 caracteres para el **nombre** y 3 para la **extensión**)
- La extensión es usada por el SO para identificar el tipo de archivo (ej: MP3, AVI, PAS, EXE, JPG).
- Windows actualmente limita a 260 caracteres, incluyendo el camino (path) y el nombre. No se pueden usar los símbolos **\\ ? : \* " > < |**
- Ej: **C:\usuarios\ale\RPA\clase-1(ale).pru.num.s.pas**

Labels: camino o ruta, nombre, extensión

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 9

### CONCEPTOS: Valores de variables en Pascal

- Muchas veces es útil que determinados **valores puedan perdurar** aunque el programa termine, y que estos valores  **puedan ser utilizados** en el futuro por **otro programa**.
- En Pascal existe un **tipo de dato estructurado FILE** (archivo/fichero) que permite almacenar valores que pueden perdurar aún cuando la ejecución del programa termine.
- De esta manera, un programa podrá "leer" elementos generados por otro programa; y además "escribir" datos que podrá leer otro programa (o él mismo pero en otra ejecución posterior).

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 10

### Declaración de archivos secuenciales en Pascal

- La palabra reservada **FILE** permite declarar un tipo o una variable de tipo archivo.

Program ejemplo;

```
VAR numeros, valores: FILE OF integer;
    caracteres: FILE OF char;
    temperaturas FILE OF real;
    datos: FILE OF boolean;
```

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 11

### CONCEPTOS: Archivos (Files)

- Para que el contenido de un archivo (file) perdure más allá de la ejecución de un programa y aún cuando la computadora estuviera apagada por un tiempo, los archivos (files) **residen en memoria secundaria** (como un disco rígido por ejemplo).
- Es importante notar que el acceso a memoria secundaria **depende del Sistema Operativo** usado.
- El manejo archivos en Pascal también puede tener **diferencias de un compilador a otro**.
- En esta materia **se verán algunos conceptos de archivos secuenciales** y algunos detalles estarán ligados al sistema operativo o al compilador.

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 12

El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:  
 "Resolución de Problemas y Algoritmos. Notas de Clase". Alejandro J. García. Universidad Nacional del Sur. (c) 2014

### CONCEPTOS: Archivos secuenciales

- Un Sistema Operativo (SO) puede manejar distintas clases de archivos (de texto, fotos, películas, música, ejecutables).
- Cada Lenguaje de Programación (LP) pueden manejar algunas de estas clases de archivos.
- En Pascal hay una clase de archivo que se denomina **archivo secuencial (FILE)**.
- Un archivo secuencial es una estructura compuesta por una secuencia de elementos (componentes) donde hay un orden lineal entre ellos.

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    13

### Conceptos:

**Tipo de Dato:** define el conjunto de valores posibles que puede tomar una variable, y también define las operaciones que pueden usarse.

Los tipos de datos en Pascal pueden ser:

- **Simples:** INTEGER, REAL, CHAR, BOOLEAN, y otros que no veremos en RPA.
- **Estructurados:** FILE (archivo), TEXT (archivo de texto) y otros que no veremos en RPA.

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    14

### Archivos secuenciales en Pascal (FILE OF ...)

- En Pascal, un archivo secuencial (FILE) es una sucesión finita de componentes que pueden accederse una a una, comenzando de la primera.
- Todos los componentes deben ser del mismo tipo de datos (ej. todos integer, o todos char).
- Por ser un archivo secuencial, una vez accedida la primera componente, el acceso a la componente de la posición P se logra luego de haber accedido previamente a la posición P-1.
- La cantidad de componentes es potencialmente infinita (su límite estará dado por la cantidad de espacio en la computadora donde está el archivo).
- Son almacenados en Memoria Secundaria.

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    15

### Representación gráfica

**Program ejemplo;**  
**VAR F1: FILE OF integer;**

- Dado que una variable de tipo FILE es una sucesión de componentes del mismo tipo, es usual usar la siguiente **representación gráfica:**

Primer elemento.

→

F1:

11	-2	5	-12	4
----	----	---	-----	---

←

Último elemento.

En F1 cada componente es de tipo integer.

- Aunque la capacidad de un archivo es potencialmente infinita, en cualquier momento dado, el archivo tendrá un número finito de componentes.

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    16

### Nombres de archivos secuenciales en Pascal

- El identificador de una variable de tipo FILE es un nombre interno usado por el programa en Pascal para referirse a un archivo secuencial.
- Como los valores almacenados en variables de tipo **FILE** van a estar en memoria secundaria (por ejemplo: disco rígido), el sistema operativo necesita asignarle un nombre válido en ese sistema.
- Este archivo puede ser usado en el futuro por otro programa usando ese nombre dado en el SO.
- La primitiva ASSIGN permite vincular el nombre interno (identificador de variable) y el nombre del archivo en el SO.

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    17

### Primitiva ASSIGN

**Program ejemplo;**  
**VAR numeros: FILE OF integer;**  
**Begin**  
**ASSIGN(numeros, 'mis-numeros.datos');**  
 ...

- La primitiva **ASSIGN(F, N)** tiene dos parámetros: **F** que es un identificador de variable de tipo FILE, y **N** que es una secuencia de caracteres que representa un nombre válido de archivo en el sistema operativo.
- El identificador **F** es llamado **manejador del archivo** de nombre **N** (*file handler*), y en el código fuente toda otra referencia al archivo se hace usando el manejador **F**.
- Una vez ejecutada **assign** vincula a **F** con **N** (el nombre real del archivo en memoria secundaria).

Resolución de Problemas y Algoritmos    Dr. Alejandro J. García    18

El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:  
 “Resolución de Problemas y Algoritmos. Notas de Clase”. Alejandro J. García. Universidad Nacional del Sur. (c) 2014

**Primitivas de Pascal para archivos secuenciales**

- **assign(F,N):** vincula F con N (nombre del archivo en SO).
- **rewrite(F):** crea un archivo nuevo con el nombre N que está vinculado a F (si ya existe otro archivo con ese nombre N se sobre-escribe y se pierde el viejo archivo).
- **write(F,e):** en un archivo F creado con rewrite, escribe el valor de "e" a continuación del último elemento de F.
- **close(F):** cierra el archivo vinculado al *manejador* F.
- **reset(F):** abre un archivo existente de nombre N para leer, y queda preparado para leer el primer elemento.
- **read(F,e):** lee un elemento del archivo F, copia el valor leído en "e" y queda preparado para leer el siguiente elemento (si existe) o queda en el fin del archivo.
- **eof(F)** (end of file – fin de archivo): retorna TRUE si se llegó al final de un archivo o FALSE en caso contrario. 19

**Problema:** escriba un programa para crear un archivo llamado "mis-numeros.dat" y escribir en él enteros de 1 a un tope ingresado por el usuario.

**Algoritmo**

Crear el archivo "mis-numeros.dat" para escribir en él.  
 Solicitar un entero tope  
 Escribir en el archivo "tope" enteros desde el 1 hasta tope  
 Cerrar el archivo

fin.

**Observación:** En este algoritmo conozco de antemano cuanto elementos quiero escribir en el archivo, entonces en Pascal puedo usar FOR.

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 20

**Problema:** escriba un programa para crear un archivo llamado "mis-numeros.dat" y escribir en él enteros de 1 a un tope ingresado por el usuario.

**Program crear;**  
**VAR** nuevo: FILE OF integer;  
 valor,tope: integer;  
**begin**  
 assign(nuevo, 'mis-numeros.dat');  
 rewrite(nuevo); *{crea archivo vacío y permite escribir en él}*  
 writeln('Cantidad de enteros a escribir en el archivo ');  
 readln(tope);  
**for** valor:= 1 **to** tope **do** write(nuevo,valor);  
 close(nuevo);  
**end.**

**Escribe un entero al final del archivo**

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 21

**Problema:** escriba un programa para abrir un archivo ya existente llamado "mis-numeros.dat", leer todos sus componentes, y mostrarlos por pantalla.

**Algoritmo**

Abrir el archivo "mis-numeros.dat" para leer sus elementos  
 Leer uno a uno los elementos y mostrarlos en pantalla  
 Cerrar el archivo.

fin.

**MUY IMPORTANTE:** si la primitiva "read(F,E)" es usada sobre el fin de un archivo (o un archivo vacío) es considerado un error de programación ya que dará un error y el programa dejará de ejecutarse. Por lo tanto antes de usar "read(F,e)" debe asegurarse no estar al final del archivo con eof(F).

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 22

**Problema:** escriba un programa para abrir un archivo ya existente llamado "mis-numeros.dat", leer todos sus componentes, y mostrarlos por pantalla.

**Program leer;**  
**VAR** arch\_num: FILE OF integer; elemento: integer;  
**begin**  
 assign(arch\_num, 'mis-numeros.dat');  
 reset(arch\_num); *{abre el archivo para leer de él}*  
**while** not eof(arch\_num) **do** *{mientras no llegue al fin}*  
**begin**  
 read(arch\_num,elemento); writeln(' fue leído:', elemento);  
**end;**  
 close(arch\_num);  
**end.**

**Lee un elemento del archivo y queda preparado para leer el siguiente.**

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 23

**Problemas propuestos**

- **Problema:** escriba un programa que cuente **cuantos elementos** tiene el archivo "mis-numeros.dat" (ya creado y con números en él).
- **Problema:** escriba un programa que busque **cuantas veces está** el elemento E (ingresado por el usuario) en el archivo "mis-numeros.dat" (ya creado y con números en él).
- **Problema:** escriba un programa que vea si **primer elemento es igual** al **último** en el archivo "mis-numeros.dat" (ya creado y con números en él).
- **Problema:** escriba un programa que vea si los elementos del archivo "mis-numeros.dat" (ya creado y con números en él) **están ordenados de menor a mayor.**

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 24

El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:  
 "Resolución de Problemas y Algoritmos. Notas de Clase". Alejandro J. García. Universidad Nacional del Sur. (c) 2014

### Referencias

- [en.wikipedia.org/wiki/Usage\\_share\\_of\\_operating\\_systems](http://en.wikipedia.org/wiki/Usage_share_of_operating_systems)
- [1] <http://www.netapplications.com/>
- [www.w3counter.com/globalstats.php](http://www.w3counter.com/globalstats.php)

Resolución de Problemas y Algoritmos      Dr. Alejandro J. García      25

### Sistemas Operativos usados según tipo de computadora

Market share by category [edit]

Category	Source	Date	Linux based	Mac and Other Unix	In-House	Windows	Other
Desktop, laptop, netbook	Net Applications <sup>[34]</sup>	Mar-14	1.49% (Ubuntu, etc.)	7.57% (OS X)		90.94% (XP, 7, Vista, 8)	
Smartphone, tablet	StatCounter Global Stats <sup>[35]</sup>	Mar-14	46.26% (Android)	33.09% (iOS)		1.78% (WP8, RT)	18.87%
Server (web)	W3Techs <sup>[36]</sup>	Jan-14	34.62% (Debian, CentOS, RHEL)	32.48% (BSD, HP-UX, Aix, Solaris)		32.90% (WZK3, WZK8)	
Supercomputer	TOP500 <sup>[33]</sup>	Nov-13	96.4% (Custom)	2.4% (UNIX)		0.4%	0.8%
Mainframe	Gartner <sup>[31]</sup>	Dec-08	28% (SLES, RHEL)				72% (z/OS)
Gaming console	Nintendo, Sony, Microsoft, Ouya <sup>[37]</sup>	Jun-13	0% (SteamOS, Android)	28.6% (PS3)	40.9% (Wii)	29.5% (Xbox)	
Embedded	UBM Electronics <sup>[38]</sup>	Mar-12	29.44% (Android, Other)	4.29% (QNX)	13.5%	11.65% (WCE 7)	41.1%
Real time	NewTechPress <sup>[39]</sup>	Nov-11	19.3% (Android)		20.1%	35.8% (XPE, WCE)	24.8%

[http://en.wikipedia.org/wiki/Usage\\_share\\_of\\_operating\\_systems](http://en.wikipedia.org/wiki/Usage_share_of_operating_systems)  
 Resolución de Problemas y Algoritmos      Dr. Alejandro J. García      26

### Sistemas Operativos – Estadísticas (web servers)

Operating Systems 2012			Operating Systems 2013		
1	Windows 7	39.47%	1	Windows 7	43.69%
2	Windows XP	29.38%	2	Windows XP	22.59%
3	Apple OS X	8.79%	3	iOS	9.50%
4	Windows Vista	7.77%	4	Apple OS X	8.45%
5	Apple iOS	5.25%	5	Windows Vista	5.28%
6	Android	1.78%	6	Android	4.19%
7	Linux	1.76%	7	Windows 8	2.72%
8	BlackBerry	0.57%	8	Linux	1.89%
9	SymbianOS	0.18%	9	BlackBerry	0.57%
10	Windows 8	0.08%	10	SymbianOS	0.26%

<http://www.w3counter.com/globalstats.php>  
 Resolución de Problemas y Algoritmos      Dr. Alejandro J. García      27

El uso total o parcial de este material está permitido siempre que se haga mención explícita de su fuente:  
 “Resolución de Problemas y Algoritmos. Notas de Clase”. Alejandro J. García. Universidad Nacional del Sur. (c) 2014