

## Resolución de Problemas y Algoritmos

**Clase 9:**  
**Algoritmos para resolución de problemas con secuencias de datos.**



Blaise 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

### Problema propuesto: (dígito presente)

**Problema:** Escriba un programa en Pascal para determinar si un dígito D aparece en un número entero N.

Ejemplos significativos (para luego casos de prueba)

Ej: si D = 3 y N=1234, entonces D aparece.

Ej: si D = 0 y N=0, entonces D aparece.

Ej: si D = 3 y N=661, entonces D no aparece.

Observe: un número es una secuencia de dígitos, por lo tanto hay una analogía con el problema que ya fue resuelto antes de buscar un elemento en una secuencia ingresada por pantalla.

**Solución:** voy mirando dígito a dígito en N, si alguno de los dígitos de N es igual a D es que aparece; si recorro todos los dígitos de N y ninguno es igual a D entonces no aparece.

¿Cómo puedo "mirar" un dígito particular de un número N?

→ Para obtener el último dígito de N puedo hacer : **N mod 10**

→ Para eliminar el último dígito de N puedo hacer: **N div 10**

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 2

### Algoritmo para dígito presente en un número

**Solución:** voy mirando dígito a dígito y si alguno es igual a D es que aparece; si recorro todo el número y ninguno es igual a D entonces no aparece.

**Observación:** un número siempre tiene al menos un dígito (se repetirá 1 o más veces)

**Algoritmo:**

**Obtener los datos N y D**

**Repetir**

Si el último dígito de N es igual a D ( $N \bmod 10 = D$ )  
entonces D aparece en N

Eliminar el último dígito de N ( $N \leftarrow N \div 10$ )  
hasta que: "N no tenga mas dígitos" o "descubrí que aparece"

Si aparece entonces mostrar "D aparece en el número"  
de lo contrario mostrar "D no aparece en el número"

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 3

```
PROGRAM digito_presente; {indica si está un dígito en un número}
VAR numero, digito, cant_veces : INTEGER; esta : BOOLEAN;
BEGIN
writeln('ingrese Num y Dig'); readln(numero, digito);
esta:=false; {asumo que no está}
REPEAT {Todo número tiene 1 o más dígitos}
  IF (numero mod 10) = digito {comparo con el último dígito}
  THEN esta:=true
  ELSE numero := numero div 10; {elimino el último dígito}
  // writeln(numero); descomentar para ver como cambia numero
UNTIL (numero = 0) OR esta; {hasta recorrer todo o lo encontré}
IF esta {también puede ser "esta=true" pero es redundante}
then writeln(digito, ' aparece en el número ingresado')
else writeln(digito, ' no aparece en el número ingresado');
END.
```

Resolución de Problemas y Algoritmos Dr. Alejandro J. García 4

### Archivos (repaso)

**Program ejemplo;**  
**VAR arch\_reales: FILE OF real;**

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

Primer elemento.

Último elemento.

Toda componente es de tipo integer.

Arch\_reales: 1.9 -2.1 0.05 -12

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 5

### Primitivas de Pascal para archivos secuenciales

**VAR F: FILE OF <tipo base>; e: <tipo base>;**

- **assign(F,N):** vincula el manejador de archivo F con el nombre **N** que usará el archivo en el Sistema Operativo.
- **rewrite(F):** crea un archivo nuevo con el nombre **N** vinculado al manejador **F** (si ya existe otro con nombre **N** lo sobre-escribe).
- **write(F,e):** en el archivo asociado a **F** y 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 para leer un archivo existente vinculado al manejador F, y queda preparado para leer el primer elemento de **F**.
- **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 asociado a **F**.
- **eof(F)** (end of file – fin de archivo): retorna **TRUE** si se llegó al final de un archivo o **FALSE** en caso contrario.

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) 2015**

**Problema:** crear un archivo de números reales llamado "numeros.reales" y permitir al usuario cargar elementos en el archivo.

**Algoritmo**

Crear el archivo.  
 Solicitar la cantidad de archivos.  
 Repetir cantidad veces:  
     Leer un elemento,  
     Escribir en el final del archivo  
 Cerrar el archivo.

fin.

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

**Una posible solución**

```

Program Crear; {crea y carga un archivo de reales}
VAR numeros: FILE OF real; i,cantidad: integer; e:real;
begin
    assign(numeros, 'numeros.reales');
    rewrite(numeros);
    writeln('Cantidad'); readln(cantidad);
    for i:=1 to cantidad do begin
        write(' Elemento :'); readln(e); {leo del buffer}
        write(numeros,e); {guardo al final del archivo}
    end;
    close(numeros);
    writeln('archivo creado'); readln;
end.
    
```

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

**Problema:** Indicar si un valor particular ingresado por el usuario está presente en un archivo de números reales llamado "numeros.reales"

**Solución:** observe que existe una analogía con "dígito presente" que puede aprovecharse (aún cuando su enunciado es diferente)

Ir comparando elemento a elemento y si alguno es igual al buscado está presente, pero si recorro todo el archivo y ninguno es igual al buscado entonces no está presente.

**Algoritmo pertenece al archivo**

Abrir el archivo para leer y solicitar el elemento a buscar.  
 Repetir mientras no sea fin de archivo y no encontré al buscado leer un elemento del archivo  
     si elemento leído = buscado entonces encontré el buscado  
 Cerrar el archivo.  
 Indicar si fue encontrado

fin.

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

**Una posible solución**

```

Program buscar; {busca un elemento en un archivo}
VAR numeros:FILE OF real; encontrado:boolean; buscado, e:real;
begin
    assign(numeros, 'numeros.reales'); reset(numeros);
    write('? Buscado?'); readln(buscado); encontrado:=false;
    while not eof(numeros) and not encontrado do begin
        read(numeros,e);
        if e = buscado then encontrado:=true;
    end;
    close(numeros);
    if encontrado then writeln('Elemento encontrado')
    else writeln('El elemento no fue encontrado');
end.
    
```

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

**Problema:** mostrar todos los elementos de un archivo de números reales llamado "numeros.reales" y además indicar cuantos elementos fueron mostrados. Por ejemplo, si el archivo tiene los elementos: 11, -2 y 5, deberá mostrar e indicar que fueron mostrados 3 números. Si el archivo está vacío deberá indicarlo en pantalla.

**Algoritmo**

Abrir el archivo para leer.  
 Leer uno a uno los elementos, mostrarlos en pantalla, y contar los que se van mostrando.  
 Cerrar el archivo.  
 Mostrar cuantos fueron leído (o que el archivo estaba vacío).

fin.

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

**Una posible solución**

```

Program mostrar; {muestra y cuenta los elem. de un archivo}
VAR numeros: FILE OF real; cantidad: integer; e:real;
begin
    assign(numeros, 'numeros.reales');
    reset(numeros); Cantidad:=0;
    while not eof(numeros) do begin
        read(numeros,e); Cantidad:=cantidad+1;
        writeln(' Elemento leído:', e:10:3);
    end;
    close(numeros);
    if cantidad > 0 then writeln('mostrados =', cantidad)
    else writeln('el archivo estaba vacío');
end.
    
```

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) 2015

**Problema :** considere un archivo de números enteros llamado "accesos.web.marzo" (ya creado) que tiene la cantidad de accesos a una página web por día en el mes de marzo. Escriba un programa que **calcule la cantidad de accesos promedio en el mes.**

**Solución:**  
 Observe como la **experiencia** de la solución al problema anterior puede aprovecharse para resolver nuevos problemas donde, aún cuando su enunciado es diferente, presentan semejanzas (también llamada **analogía**) en su solución.

Resolución de Problemas y Algoritmos Dr. Alejandro J. García

**Una posible solución**

```

Program promedio; {promedio de los elem. de un archivo}
VAR numeros: FILE OF real; cantidad: integer; suma, e: real;
begin
  assign(numeros, 'accesos.web.marzo');
  reset(numeros); Cantidad:=0; suma:=0;
  while not eof(numeros) do begin
    read(numeros,e); Cantidad:=cantidad+1;
    suma:=suma+e;
  end;
  close(numeros);
  if cantidad > 0 then writeln('promedio =', suma/cantidad)
  else writeln('el archivo estaba vacío');
end.
    
```

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

**Problema para practicar**

**Problema :** considere un archivo de números reales llamado "temperaturas.marzo" (ya creado). Escriba un programa que **calcule la temperatura promedio.**

**Solución:**  
 Observe como la **experiencia** de la solución al problema anterior puede aprovecharse para resolver nuevos problemas donde, aún cuando su enunciado es diferente, presentan semejanzas (también llamada **analogía**) en su solución.  
 Esto puede utilizarse como una estrategia en la resolución de problemas.

Resolución de Problemas y Algoritmos Dr. Alejandro J. García

**Problema para practicar**

**Problema :** considere un archivo de nros. reales llamado "temperaturas.marzo" (ya creado). Escriba un programa que **calcule la temperatura máxima.**

Observe nuevamente como la experiencia de la solución al problema anterior puede aprovecharse para resolver problemas donde, aún cuando su enunciado es diferente, presentan semejanzas en su solución.

Resolución de Problemas y Algoritmos Dr. Alejandro J. García

**Problema:** escriba un programa para copiar el contenido de "mis-numeros.datos", a otro archivo "mis-numeros.copia".

**Program copiar; (VERSION INCORRECTA)**

```

VAR F1,F2: FILE OF integer;
begin
  assign(F1, 'mis-numeros.datos');
  assign(F2, 'mis-numeros.copia');
  F1:=F2;
  close(F2); close(F1);
end.
    
```

**MAL**

Esto no copia los archivos

Esta asignación hace que los dos manejadores se refieran al mismo archivo: "mis-numeros.copia"

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

**Problema:** escriba un programa para copiar el contenido de "mis-numeros.datos", a otro archivo "mis-numeros.copia".

**Program copiar; {version correcta}**

```

VAR F1,F2: FILE OF integer;
  elemento: integer;
begin
  assign(F1, 'mis-numeros.datos'); reset(F1);
  assign(F2, 'mis-numeros.copia'); rewrite(F2);
  while not eof(F1) do begin
    read(F1,elemento);
    write(F2, elemento);
  end;
  close(F1); close(F2);
end.
    
```

Lee un elemento del archivo F1 y luego escribe elemento en F2

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) 2015

### Blaise Pascal (1623 -1662)

Matemático, físico, filósofo y teólogo francés, considerado uno de los padres de las computadoras junto con *Charles Babbage*.

Desde los 12 se dedicó a las matemáticas, cuando tenía 19 años comenzó a trabajar en su calculadora mecánica (llamada Pascaline o Arithmetique).

Luego se dedicó al física en especial la presión atmosférica y posteriormente a la filosofía.

Murió en París a los 39 años.

[http://es.wikipedia.org/wiki/Blaise\\_Pascal](http://es.wikipedia.org/wiki/Blaise_Pascal)



### Roue Pascaline

En 1640, su padre fue nombrado Comisario Real y jefe de recaudación de impuestos para la Normandía con asiento en Ruan.

Allí, en 1642, Pascal inventó la Roue Pascaline, «Rueda de Pascal» o Pascalina. Pretendía ayudar a su padre buscando crear un dispositivo que pudiera reducir un poco su carga de trabajo.

Inicialmente solo permitía realizar adiciones, pero recibió permanentes mejoras, siendo finalmente capaz de realizar restas. Pascal la hizo patentar, pero no se cumplieron sus expectativas de hacerse rico. Las máquinas, confeccionadas una a una y a mano, eran demasiado caras y solo llegó a fabricar 50, de las que subsisten nueve.



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) 2015