

Septiembre 2003

Se desea realizar un programa en *Perl* para la detección de mensajes de correo electrónico potencialmente peligrosos, por ser susceptibles de distribuir un virus informático. El programa debe ser capaz de detectar los mensajes que contienen palabras que incluyan 3 vocales consecutivas. En caso de detectar ese tipo de palabras en un *email*, el programa debe indicar el remitente y el número de palabras de este tipo localizadas en el mensaje.

Para realizar el programa hay que saber que un mensaje de correo electrónico es un fichero texto, formado por unas líneas de cabecera y un cuerpo con el contenido del mensaje. Las líneas de cabecera siempre tienen el mismo formato:

<etiqueta>: <texto>

Ejemplos de líneas de cabecera:

De: Jesús Albert

Para: webmaster@perl.org

Asunto: Pregunta de examen

El número de líneas que componen la cabecera es variable y dependen del *email*. El cuerpo con el texto del mensaje se inicia inmediatamente después de la última línea de cabecera.

Finalmente, hay que tener en cuenta que las palabras *prohibidas* deben aparecer en el cuerpo del mensaje, no en las líneas de cabecera y el remitente del correo está indicado en la línea de cabecera con la etiqueta "De" (que siempre existe).



Septiembre 03, solución (I)

```
#!/usr/bin/perl
use strict;
use warnings;

##Comprobar argumentos
if ($#ARGV != 0)
{
    die "Uso: $0 <email>\n";
}

my $email = $ARGV[0];
##Abrir archivo
open (IN, $email) || die ("Error al abrir $email\n");
```



Septiembre 03, solución (II)

```
##Leer cabecera y guardar el remitente
my $linea;
my $remitente;
while (defined($linea=<IN>)&&($linea=~/(.+):\s+(.*))
{
    if ($1 eq "De")
    {
        $remitente = $2;
        chomp($remitente);
    }
}
```



Septiembre 03, solución (III)

```
##Leer el mensaje y contar apariciones del patrón
if (defined ($linea))
{
    do
    {
        if ($linea =~ /[aeiouAEIUO]{3,}/)
        {
            @lista = split (/s+/, $linea);
            foreach my $i (@lista)
            {
                if ($i =~ /[aeiouAEIOU]{3,}/)
                {
                    $cont ++;
                }
            }
        }
    }
}

while (defined ($linea = <IN>))
```



Septiembre 03, solución (IV)

```
##Si el numero> 0 entonces imprimir remitente y numero
##sino, indicar que no hay nada

if ($cont > 0)
{
    print "Posible virus remitido por $remitente,
$cont apariciones de patron prohibido.\n";
}

else
{
    print "Mensaje seguro.\n";
}
```



Examen Junio 2003

Tenemos tres ficheros que contienen información sobre los alumnos matriculados en la asignatura de Herramientas de Programación: el fichero "matriculas.dat" contiene el nombre de los alumnos (nombre y apellidos) y su código de identificación; el fichero "notas.dat" contiene el código de identificación del alumno y siete calificaciones del mismo (de 0 a 10), las cinco primeras son notas de laboratorio y las dos últimas son las notas de dos exámenes realizados (nota de teoría). La nota final de cada alumno es el 20% de la nota de laboratorio y el 80% la nota de teoría. Por último, el fichero "convocatoria.dat" que contiene el código de identificación del alumno y la convocatoria en la que se encuentra.

Se pide, escribir un programa en Perl que realice unas estadísticas sobre las notas obtenidas por los alumnos, indicándonos los porcentajes de aprobados y suspendidos por convocatoria. Los resultados de estas estadísticas se escribirán en un fichero. La tabla estadística que se escribirá debe ser como la siguiente:

	1	2	3	4	5	6
Aprobados %						
Suspendidos %						

También queremos saber el alumno que ha obtenido la nota más alta y el que ha obtenido la nota más baja. De estos dos alumnos deseamos conocer sus nombres completos, su código de identificación y en qué convocatoria se encuentra.



Examen Junio 2003 (II)

Ayuda: Ejemplos de los contenidos de los ficheros.

"matriculas.dat"

```
Jesús Albert Blanco 40220
Vicente Caverio Garcia 89898
Xaro Benavent Garcia 43423
Juan de Mata Domingo Esteve 5555
```

"notas.dat"

```
89898 1.2 8.9 2.3 4.2 4.2 5.2 2.1
40220 5.6 5.7 8.2 4.2 7.8 8.2 9.5
5555 2.5 2.3 4.2 5.2 5.2 5.3 7.8
43423 8.5 6.7 8.2 9.2 5.6 6.7 8.9
```

"convocatoria.dat"

```
89898 5
43423 2
40220 1
5555 2
```



Junio, solucion (I)

```
#!/usr/bin/perl

use strict;
use warnings;

#Obrim els tres fitxers on tenim la informació
emmagatzemada
open(FMAT, "matricules.dat") || die("No podem obrir
fitxer matricules.dat");
open(FNOTES, "notes.dat") || die("No podem obrir
fitxer notes.dat");
open(FCONVOC, "convocatoria.dat") || die("No
obrir el fitxer convocatoria.dat");
```



Junio, solució (II)

```
#Usem tres taules hash per a emmagatzemar la
informació
# La clau per a totes les taules hash és el número
de matrícula
# La taula noms guarda {num_mat,nom_alumne}
# La taula notes guarda {num_mat,notafinal)
# La taula convocatòria guarda {num_mat,convocatòria}

my %noms;
my %notes;
my %convocatòria;
my $line;
```



Junio, solució (III)

```
#Omplim la taula hash %noms
while (defined($line= <FMAT>))
{
    if ($line =~ /[a-zA-ZàéíóúÁÉÍÓÚ\ ]*([0-9]*)/)
    {
        my $nom= $1;
        my $nummat= $2;

        $noms{$nummat}=$nom;
    }
}
```



Junio, solucion (IV)

```
# Omplim la taula hash de notes (amb la nota final calculada)
# Al mateix temps que anem calculant les notes dels alumnes
# ens guardarem el que té la nota més alta i el que té la nota més baixa

my $mesnota=0.0;
my $matmesnota;
my $menosnota=10.0;
my $matmenosnota;

while (defined ($line=<FNOTES>))
{
    my @not= split("\ ",$line);
    my $nummat= $not[0];

    my $nota= (($not[1]+$not[2]+$not[3]+$not[4]+$not[5])/5.0)*0.2+((($not[6]+$not[7])/2.0)*0.8;
    $notes{$nummat}=$nota;

    if ($nota>$mesnota)
    {
        $mesnota=$nota;
        $matmesnota= $nummat;
    }
    if ($nota<$menosnota)
    {
        $menosnota=$nota;
        $matmenosnota= $nummat;
    }
}
```



Junio, solucion (V)

```
#Omplim la taula hash del número de convocatoria
while (defined ($line=<FCONVOC>))
{
    my @conv= split("\ ",$line);
    my $nummat= $conv[0];
    my $c= $conv[1];
    $convocatoria{$nummat}=$c;
}

close (FMAT);
close (FNOTES);
close (FCONVOC);
```



Junio, solucion (VI)

```
#En els arrays de aprovats i suspensos guardem
# el nombre d'aprovats i suspensos respectivament
# per convocatoria
# @aprovats=[10 3 4 4]
# 10 primera convocatoria
# 3 en la segona convocatòria, etc.

#Tenim un total de 6 convocatòries
#Hem d'inicialitzar els vectors a 0

my @aprovats;
my @suspensos;
my @totals;
#El vector comencem en 1 perquè no tenim convocatòria 0
for (my $i=1;$i<=6;$i=$i+1)
{
    $aprovats[$i]=0;
    $suspensos[$i]=0;
    $totals[$i]=0;
}
```



Junio, solucion (VII)

```
#En la taula de notes tenim les notes i el número de matrícula
# dels alumnes
# Hem de comptar el nombre d'aprovats, el nombre de suspensos y el total
# i emmagatzemar-los al vector corresponent per convocatoria.
foreach my $clau (keys %notes)
{
    my $con= $convocatoria{$clau};
    if ($notes{$clau}>5.0)
    {
        $aprovats[$con]=$aprovats[$con]+1;
    }
    else
    {
        $suspensos[$con]=$suspensos[$con]+1;
    }
    $totals[$con]=$aprovats[$con]+$suspensos[$con];
}
```



Junio, solucion (VIII)

```
#Escribim els resultats a un fitxer on es emmagatzemem les
estadistiques
open(EIXIDA, ">estadist.dat")||die("No podem obrir fitxer
estadist.dat");
print EIXIDA "\t\t1\t2\t3\t4\t5\t6\n";
print EIXIDA "aprovals\t";

for (my $i=1;$i<=6;$i++)
{
    my $out;
    if ($totals[$i] == 0)
    { $out= 0}
    else
    { $out=($aprovals[$i]/$totals[$i])*100.0;}

    print EIXIDA "$out\t";
}
```



Junio, solucion (IX)

```
print EIXIDA "\nsuspensos\t";
for (my $i=1;$i<=6;$i++)
{
    my $out;
    if ($totals[$i] == 0)
    { $out=0;}
    else
    { $out=($suspensos[$i]/$totals[$i])*100.0;}
    print EIXIDA "$out\t";
}

print EIXIDA "\nLa nota mes alta:\n $matmesnota $noms{$matmesnota}
$notes{$matmesnota} $convocatoria{$matmesnota}\n";
print EIXIDA "La nota mes baixeta:\n $matmenosnota $noms{$matmenosnota}
$notes{$matmenosnota} $convocatoria{$matmenosnota}\n";
print EIXIDA "\n";
close (EIXIDA);
```

