

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL  
 INSTITUTO DE CIENCIAS MATEMÁTICAS  
 SEGUNDA EVALUACIÓN  
 I TÉRMINO 2009-2010

Matrícula: \_\_\_\_\_ Nombre: Luis Rodríguez Ojeda Paralelo: \_\_\_\_\_

**Tema 1 (10 puntos).** Realice la función **desordena(palabra)**, la cual recibe una palabra y desordena las letras en forma aleatoria. Ejemplo:

>> desordena('roma') ans='mora'

Sugerencia: Para cada letra, seleccione aleatoriamente otra letra de la palabra con la que intercambiarán posiciones.

Rúbrica: Planteo sintáctico de la función (3 puntos), intercambio aleatorio (7 puntos)

```
function p=desordena(x)
```

```
p=[];
```

```
while length(x)>0
```

```
    i=fix(rand*length(x)+1);
```

```
    p=[p, x(i)];
```

```
    x(i)=[];
```

```
end
```

```
>> x='prueba';
```

```
>> p=desordena(x)
```

```
p =
```

```
praueb
```

```
>> p=desordena(x)
```

```
p =
```

```
repabu
```



JARAL II

F. Computación

**Tema 2 (20 puntos).** En un juego de adivinanza, un primer jugador escribe una palabraA, se desordenan sus letras, y se muestran al otro jugador (usando la función del tema anterior). El jugador que desea adivinar, usando las letras mostradas, escribe otra palabraB. Si es igual a palabraA gana el juego, sino, tiene tantas oportunidades como letras haya en la palabra.

- a) Realice la función **equivocadas(palabraA, palabraB)**, la cual compara dos palabras de igual tamaño e indica cuantas letras no son iguales.
- b) Presente un programa para realizar el juego propuesto y que utilice las funciones descritas anteriormente.

**Ejemplo:**

**palabraA: roma**

**desordenada: oamr**

**adivina: omar**

**adivina: ramo**

**adivina: roma**

**adivinaste!!!!**

*Sugerencia: Se supone que la función del tema anterior está completa y correcta*

*Rúbrica: Función "equivocadas" (8 puntos), programa del juego y uso de funciones (12 puntos)*

```
function c=equivocadas(a, b)
c=0;
for i=1:length(a)
    if a(i)~=b(i)
        c=c+1;
    end
end

a=input('ingrese una palabra ');
b=desordena(a);
disp('Palabra desordenada');
disp(b)
for i=1:length(a)
    x=input('Adivine la palabra ');
    c=equivocadas(a,x);
    if c==0
        break;
    end
end
if c==0
    disp('Respuesta correcta');
else
    disp('Respuestas incorrectas');
end
```

```
>> pr
ingrese una palabra 'prueba'
Palabra desordenada
aerupb
Adivine la palabra 'rupbae'
Adivine la palabra 'prueba'
Respuesta correcta
```



**Tema 3 (35 puntos).** Para realizar el monitoreo del trabajo de ventas, una empresa registra las unidades vendidas de cada vendedor por producto.

Los mejores vendedores por categoría son aquellos que:

- a) Venden más unidades
- b) Su monto total (en dólares) por ventas es el mayor

Vendedor	VENTAS	Producto			
		1	2	3	4
1	10	0	12	7	
2	6	4	13	19	
3	2	17	8	9	
4	4	19	16	13	
5	9	14	14	11	
n	...	...	...	...	

Lista de Precios	
Producto	Precio
1	10
2	20
3	30
4	40
m	...

Realice un **programa** que:

- Registre las ventas mensuales de **n** vendedores por **m** productos y la lista de precios para los **m** productos, luego encuentre los mejores vendedores por cada categoría planteada.

Monto = (Unidades vendidas de un producto) \* (precio de un producto)

Rúbrica: Registro de datos (5 puntos), mejor vendedor/unidades (15 puntos), mejor vendedor/monto (15 puntos)

```

v=input('Ingrese datos de ventas ');
[n, m]=size(v);
p=input('Ingrese la lista de precios ');
for i=1:m
    t(i)=0;
    for j=1:n
        t(i)=t(i)+v(i,j);
    end
    d(i)=t(i)*p(i);
end
maxventa=max(t);
maxmonto=max(d);
disp('Mejores vendedores por cantidad');
for i=1:m
    if t(i)==maxventa
        disp(i);
    end
end
disp('Mejores vendedores por monto');
for i=1:m
    if t(i)*p(i)==maxmonto
        disp(i);
    end
end
end
    
```

*blog.espol.edu.ec/eduCasanas/*

```

>> pr
Ingrese datos de ventas [2 3 4; 5 6 7; 8 9 10]
Ingrese la lista de precios [1 2 3]
Mejores vendedores por cantidad
3
Mejores vendedores por monto
3
    
```



**Tema 4 (35 puntos).** Para llevar el control en el país de la gripe AH1N1, se requiere disponer de los registros de pacientes contagiados, potenciales y descartados (codificados como 2, 1, 0 respectivamente). El registro se realiza en cada hospital o centro de salud con los siguientes datos y se administra en el siguiente menú:

Registro de Pacientes			
Nombre	Provincia	Tipo	Hospital
María	Guayas	1	León Becerra
Juan	Manabí	2	General Manta
Pedro	Manabí	0	General Manta
José	Pichincha	0	Eugenio Espejo
Carlos	Pichincha	2	Eugenio Espejo

**Menú**

- 1.- Registrar datos de paciente
- 2.- Lista con Total de pacientes por tipo
- 3.- Total de pacientes por provincia
- 4.- Salir

Realice un programa que permita realizar las acciones planteadas en el menú

Rúbrica: Definición de estructura o archivo y uso en el programa (10 puntos), realización de menú (10 puntos), conteo de pacientes por tipo(5), por provincia (10).

```

global r;
r=[];
op=0;
while op~=4
    disp('1) Registrar');
    disp('2) Lista por tipo');
    disp('3) Lista por provincia');
    disp('4) Salir');
    op=input('Elija una opcion ');
    switch op
        case 1, registrar;
        case 2, listart;
        case 3, listarp;
    end
end

function registrar %Ingreso de datos
global r;
x.n=input('Ingrese nombre ');
x.p=input('Ingrese provincia (numero) ');
x.t=input('Ingrese tipo (0, 1, 2) ');
x.h=input('Ingrese hospital ');
r=[r, x];
    
```



```
function listart          %Pacientes por tipo
global r;
c=zeros(3,1);
for i=1: length(r)
    k=r(i).t + 1;
    c(k)=c(k) + 1;
end
disp('Pacientes por tipo');
for k=1:3
    fprintf('Tipo %d Total %d\n',k-1,c(k));
end
```

```
function listarp          %Pacientes por provincia
global r;
c=zeros(24,1);
for i=1: length(r)
    k=r(i).p;
    c(k)=c(k)+1;
end
disp('Pacientes por provincia');
for k=1:24
    fprintf('Provincia %d Total %d\n',k,c(k));
end
```



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL  
FACULTAD DE CIENCIAS NATURALES Y MATEMÁTICAS  
DEPARTAMENTO DE MATEMÁTICAS  
2da Evaluación de FUNDAMENTOS DE COMPUTACIÓN  
11 de ABRIL de 2013

Tema 1 (30 puntos)

Los números de Eudoxus se definen como sigue:

$$\begin{cases} x_r = y_r + y_{r-1}, & r \geq 1 \\ y_r = x_{r-1} + y_{r-1}, & r \geq 1 \\ x_0 = 1 \\ y_0 = 1 \end{cases}$$

Realice las funciones `eudoxusX(r)` y `eudoxusY(r)`, y un programa que genere la secuencia de pares ordenados  $(x_r, y_r)$ , llamando a estas funciones, para  $0 \leq r \leq n$ . El valor de  $n$  será leído por teclado.

```
function x=eudoxusX(r)
if r>=1
    x=eudoxusY(r)+eudoxusY(r-1);
else
    x=1;
end
```

```
function y=eudoxusY(r)
if r>=1
    y=eudoxusX(r-1)+eudoxusY(r-1);
else
    y=1;
end
```

```
% Programa prueba
n=input('Ingrese n ');
for r=0:n
    x=eudoxusX(r);
    y=eudoxusY(r);
    disp([x,y]);
end
```

```
>> prueba
Ingrese n 6
    1    1
    3    2
    7    5
   17   12
   41   29
   99   70
  239  169
```

*blog.espol.edu.ec/educasanas/*



## Tema 2 (35 puntos)

Desarrolle una aplicación para poder administrar los registros de personas que asisten a dos conferencias (1 o 2). Los registros se realizarán ingresando el código de cada persona con el siguiente menú:

- 1) Registrar
- 2) Consultar
- 3) Eliminar
- 4) Salir

En la opción 1, Elija la conferencia, ingrese el código, verifique que no está registrado y agregue al vector respectivo.

En la opción 2, Ingrese el código, busque en qué conferencia está registrado y muestre un mensaje apropiado.

En la opción 3, Ingrese el código, verifique si está registrado y elimínelo del vector respectivo.

Use dos vectores para almacenar los códigos registrados en las dos conferencias. Declare los vectores como variables globales y use funciones para instrumentar las acciones que correspondan a cada opción.

```
function menu1
disp('1) Registrar');
disp('2) Consultar');
disp('3) Eliminar');
disp('4) Salir');
```

```
function registrar
global c1 c2
c=input('Elija la conferencia ');
if c==1 | c==2
k=input('Ingrese el código ');
if c==1
e=ismember(k,c1);
if e==0
c1=[c1, k];
end
else
e=ismember(k,c2);
if e==0
c2=[c2, k];
end
end
if e==1
disp('Código ya existe');
end
else
disp('Error en el número de conferencia');
end
```



```

function consultar
global c1 c2
k=input('Ingrese el código ');
e1=ismember(k,c1);
if e1==1
    disp('Registrado en la conferencia 1');
end
e2=ismember(k,c2);
if e2==1
    disp('Registrado en la conferencia 2');
end
if e1==0 & e2==0
    disp('No está registrado');
end

```

```

function eliminar
global c1 c2
k=input('Ingrese el código ');
[e1,p]=ismember(k,c1);
if e1==1
    c1(p)=[];
end
[e2,p]=ismember(k,c2);
if e2==1
    c2(p)=[];
end
if e1==0 & e2==0
    disp('No está registrado');
end

```

```

%Programa prueba
global c1 c2;
c1=[];
c2=[];
opc=0;
while opc~=4
    menu1;
    opc=input('Elija una opción ');
    switch opc
        case 1, registrar;
        case 2, consultar;
        case 3, eliminar;
    end
end

```



### Tema 3 (35 puntos)

Considere que para cada proyecto de cierta empresa se tienen los siguientes datos: un código único (número secuencial entero), nombre del proyecto, presupuesto en dólares, tiempo de duración en semanas y prioridad (tipos 1, 2 o 3).

Realice un programa, utilizando estructuras, para administrar  $n$  proyectos con: el código (numerado desde 1), el nombre (máximo de 30 caracteres), el presupuesto (valor real positivo), el tiempo de duración (valor entero positivo) y la prioridad. El Gerente de esta empresa desea conocer una lista con los proyectos cuyo nombre empieza con la letra A y cuyo presupuesto sea mayor al promedio del presupuesto de todos los proyectos.

```
%Programa prueba
```

```
n=input('Cantidad de proyectos ');
```

```
s=0;
```

```
for i=1:n
```

```
    p(i).cod=input('Ingrese código ');
```

```
    p(i).nom=input('Nombre ','S');
```

```
    p(i).pre=input('Presupuesto ');
```

```
    p(i).dur=input('Duración ');
```

```
    p(i).pri=input('Prioridad ');
```

```
    s=s + p(i).pre;
```

```
end
```

```
prom=s/n;
```

```
for i=1:n
```

```
    if p(i).pre>prom & p(i).nom(1)=='A'
```

```
        disp(p(i).cod);
```

```
        disp(p(i).nom);
```

```
        disp(p(i).pre);
```

```
    end
```

```
end
```





Matrícula: ..... Nombre: ..... Paralelo: ..... Firma: .....

**Tema 1 (25 puntos) El "Juego Lucky"**

- Utiliza **cuatro fichas rojas ('R')**, una **ficha azul ('A')** y un **dado**.
- Se cubren cada ficha con un vaso y aleatoriamente se alinean los vasos.
- Al reemplazar la ficha azul por el valor del dado y las rojas por 0 se formará un **número equivalente**.
- El jugador puede realizar dos cambios de posición entre los vasos sin levantarlos.
- Al cambiar las posiciones se informa al jugador si el **número equivalente**: aumentó, es igual o disminuyó.
- Al final, se descubren las fichas mostrando los puntos logrados, formado por el **número equivalente**.



- Realice una función **puntos(cadena,valor)** que reciba una cadena de caracteres entre 'R' y 'A', reemplace la letra 'R' por 0 y la letra 'A' por **valor**; entregue el equivalente numérico de puntos obtenidos.
- Escriba un programa que simule el juego preguntando cuántas rondas se jugarán, al final mostrará los puntos acumulados por el jugador. Nota: Utilice una cadena de caracteres para simular las fichas y sus posiciones.

```
Ejemplo:
>> puntos('RRRAR',5)    ans = 50
>> puntos('RARRR',5)    ans = 5000
```

Rubrica: Definición y uso de función (10 puntos). Cambio de posiciones (5 puntos). Mostrar cambios de valor (5 puntos). Programa integral (5 puntos)

```
function x=puntos(c,v)
for i=1:5
    if c(i)=='R'
        c(i)='0';
    end
    if c(i)=='A'
        c(i)=num2str(v);
    end
end
x=str2num(c);
```

Función

```
n=input('cuantas rondas');
t=0;
for i=1:n
    disp('ronda');disp(i);
    c='RRRRR';
    p=fix(rand*5)+1;
    c(p)='A';
    v=fix(rand*6)+1;
    x=puntos(c,v);
```

Cambios de posición (Opcional)

```
for j=1:2
    disp('cambio');disp(j);
    a=input('primera ficha ');
    b=input('segunda ficha ');
    z=c(a); c(a)=c(b); c(b)=z;
end
y=puntos(c,v);
if y>x
    disp('aumentó');
else
    if y<x
        disp('disminuyó');
    else
        disp('es igual');
    end
end
disp(y);
```

```
t=t+y;
end
disp('acumulado');
disp(t);
```

Programa

blog.espol.edu.ec/educasenas/



**Tema 2 (20 puntos)** Realice la función `unicos(vector)` que recibe un vector de  $n$  enteros y entrega otro vector conformado por los elementos **no repetidos**.

Considerar que el tamaño del vector resultante puede ser menor al tamaño del vector ingresado.

Rubrica: Definición de la función (5 puntos). Determinar elementos no repetidos (5 puntos). Vector sin elementos repetidos (10 puntos).

i	vector(i)
1	12345
2	64279
3	12345
4	74351
5	12345
6	67531
7	76524

j	Resultados(j)
1	12345
2	64279
3	74351
4	67531
5	76524

```
function y=unicos(x)
n=length(x);
y=[];
for i=1:n
    if ismember(x(i),y) == 0
        y=[y, x(i)];
    end
end
```

```
function e=ismember(a,b)
n=length(b);
e=0;
for i=1:n
    if a==b(i)
        e=1;
        return;
    end
end
```

No es necesario escribir esta función pues ya existe en MATLAB

Programa

Función

*blog.elpor.com*



**Tema 3 (25 puntos).** El Consejo Nacional Electoral (CNE) para iniciar un proceso de revocatoria de mandato de alcaldes, requiere en la solicitud la presentación de al menos el 10% de firmas del registro electoral. Para realizar la revisión de los datos presentados para este proceso se dispone de:

Padrón		Lista	
1	0912345678	1	0987654321
2	0987654321	2	0567896543
3	0754321234	...	...
4	0765432456	m	...
5	0567896543	...	...
...	...	...	...
n	...	...	...

- Las cédulas del padrón en un arreglo de tamaño **n**
- Las cédulas de solicitantes de la revocatoria en un arreglo lista de tamaño **m**

- Realice una función **registrados(padrón, lista)** que indique cuántas de las cédulas de la lista están registradas en el padrón. Suponga que no hay datos repetidos.
- Realice un programa que solicite el ingreso de los arreglos **padrón** y **lista**, usando las funciones **únicos()** (del tema anterior) y **registrados()**, valide los datos de la lista para informar si los datos cumplen con al menos el 10% del padrón.

```
function k=registrados(p, l)
n=length(p);
m=length(l);
k=0;
for i=1:m
    if ismember(l(i), p) == 1
        k=k+1;
    end
end
```

```
p=input('ingrese el padrón ');
l=input('ingrese la lista ');
l=unicos(l);
r=registrados(p, l);
x=length(r)/length(p)*100;
if x>=10
    disp('Si cumple con al menos el 10%');
else
    disp('No cumple con al menos el 10%');
end
```



**Tema 4** (30 puntos). Un grupo de  $n$  personas debe elegir a su representante. Será elegido si tiene al menos la mitad de los votos, caso contrario se deberá repetir la votación desde el inicio. Cada persona es identificada con un número entero entre 1 y  $n$  y cualquiera de las personas puede ser elegida.

Escriba un programa para el proceso electoral con el siguiente menú:

- **Inicializar votación:** Poner contadores y registros de  $n$  votos en cero para nueva votación.
  - **Ingresar los votos:** se ingresa el número identificador del votante y luego el identificador de su candidato.
- Validar que el votante realice un solo voto.
- **Determinar ganador:** si lo hubo, mostrar cuál fue y si la votación cumple con mayoría requerida.
  - **Mostrar estado de votación:** Muestra cantidad de votos realizados y la cantidad que aún no votan.
  - **Salir**

Sugerencia: Use un vector de  $n$  componentes para almacenar el conteo de votos de cada una de las  $n$  personas. Utilice otro vector para registrar quienes ya han votado.

```
opc=0;
n=input('Cantidad de personas ');
while opc~=5
    disp('1) Iniciar votación');
    disp('2) Ingresar los votos');
    disp('3) Determinar ganador');
    disp('4) Mostrar estado');
    disp('5) Salir');
    opc=input('Elija una opción ');
    switch opc
        case 1
            votos=zeros(n);
            x=[];
        case 2
            i=1;
            while i<=n
                v=input('Ingrese identificador del votante ');
                if ismember(v,x)==1
                    disp('Votante ya realizó la votación ');
                else
                    x=[x, v];
                    c=input('Ingrese número del candidato ');
                    if c>=1 & c<=n
                        votos(c)=votos(c)+1;
                        i=i+1;
                    else
                        disp('Voto incorrecto');
                    end
                end
            end
        case 3
            g=0;
            for i=1:n
                if votos(i)>n/2
                    g=i;
                end
            end
            if g>0
                disp('Candidato ganador');
                disp(i);
            else
                disp('No hay ganador');
            end
        case 4
```



```
disp('Cantidad de votos realizados');
disp(length(x));
disp('Cantidad que faltan votar');
f=n-length(x);
disp(f);
end
end
```





Matrícula: \_\_\_\_\_ Nombre: Luis Rodríguez Ojeda Firma: \_\_\_\_\_ Paralelo: \_\_\_\_\_

**Tema 1.- (25 %)** Escriba una función  $x=\text{orden}(t)$  que reciba una matriz  $t$  y entregue un resultado lógico 1 o 0. La función debe sumar las filas y almacenar los resultados en un vector. Debe entregar el valor 1 si los valores en este vector están en orden ascendente, caso contrario debe entregar el valor 0.  
Escriba un programa que genere repetidamente una matriz 4x4 con enteros aleatorios de una cifra. Termine cuando encuentre una matriz que cumpla la propiedad definida en la función anterior.

```
function x=orden(t)
a=sum(t');
b=sort(a);
if a==b
    x=1;
else
    x=0;
end

%Programa
x=0;
while x==0
    t=fix(rand(4,4)*10);
    x=orden(t);
end
disp(t);
```

*blog.espal.edu.ec/educasanas*

*a = [11 12 10 22]*  
*b = [1 2 3 4]*





**Tema 2.- (25 %)** Escriba una función **v=vocales(f)** que reciba una frase **f** y entregue un vector conteniendo 5 números: la cantidad de veces que ha sido utilizada cada vocal en la frase.

Escriba un programa que lea una frase, llame a la función y encuentre la vocal que ha sido utilizada más veces.

```
function v=vocales(f)
voc='aeiou';
v=[0, 0, 0, 0, 0];
n=length(f);
for i=1:n
    [e,p]=ismember(f(i),voc);
    if p>0
        v(p)=v(p)+1;
    end
end

%Programa
f=input('Ingrese una frase ','s');
v=vocales(f);
[e,p]=max(v);
voc='aeiou';
disp(voc(p));
```





**Tema 3.- (20 %)** Escriba una función recursiva `c=conteo(n)` que reciba un entero positivo `n` y entregue como resultado la cantidad de cifras que tiene el número `n`. Escriba la definición matemática de la función y luego escriba la versión en MATLAB

$$\text{conteo}(n) = \begin{cases} 1, & n < 10 \\ \text{conteo}(\text{fix}(n/10)) + 1, & n \geq 10 \end{cases}$$

```
function c=conteo(n)
if n<10
    c=1;
else
    c=conteo(fix(n/10))+1;
end
```



**Tema 4.- (30 %)** Escriba un programa con un menú para registrar estudiantes en uno de los dos paralelos de una materia mediante las opciones indicadas a continuación. Cada paralelo debe ser representado mediante un vector y cada acción debe ser instrumentada mediante una función

**1) Registrar**

Lea el numero del paralelo elegido (1 o 2), luego lea el código del estudiante y agréguelo al vector correspondiente

**2) Consultar**

Lea el código del estudiante, búsquelo en los vectores y muestre el paralelo en el que está registrado

**3) Cambiar**

Lea el código del estudiante. Si está registrado elimínelo del vector y agréguelo al otro vector

**4) Salir**

```

%Programa
global p1 p2;
clear all;
opc=0;
while opc~=4
    menu;
    opc=input('Elija una opción ');
    switch opc
        case 1, registrar;
        case 2, consultar;
        case 3, cambiar;
    end
end

function menu
disp('1) Registrar');
disp('2) Consultar');
disp('3) Cambiar');
disp('4) Salir');

function registrar
global p1 p2;
n=input('ingrese paralelo (1 o 2) ');
e=input('Ingrese código del estudiante ');
if n==1 & ismember(e,p1)==0
    p1=[p1, e];
else
    if n==2 & ismember(e,p2)==0
        p2=[p2, e];
    end
end
end
  
```



```
function consultar
global p1 p2;
e=input('Ingrese código del estudiante ');
if ismember(e,p1)==1
    disp('Paralelo 1');
else
    if ismember(e,p2)==1
        disp('Paralelo 2');
    end
end
end
```

```
function cambiar
global p1 p2;
e=input('Ingrese código del estudiante ');
[x,p]=ismember(e,p1);
if x==1
    p1(p)=[];
    p2=[p2, e];
else
    [x,p]=ismember(e,p2)
    if x==1
        p2(p)=[];
        p1=[p1, e];
    end
end
end
```





ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL  
FACULTAD DE CIENCIAS NATURALES Y MATEMÁTICAS  
DEPARTAMENTO DE MATEMÁTICAS  
ICM00794 - FUNDAMENTOS DE COMPUTACION  
2da Evaluación II TÉRMINO 2012-2013 – Enero 29, 2013



MATRICULA: ..... NOMBRE: ..... L. Rodríguez O. .... PARALELO: .....

NOTA: Este examen está diseñado para ser resuelto de manera individual, puede usar un lápiz o esferográfico. Solo puede comunicarse con la persona responsable de la recepción del examen; y, cualquier instrumento de comunicación que hubiera traído, deberá apagarlo y ponerlo en la parte anterior del aula, junto con algún otro material que se encuentre acompañándolo. No consultará: libros, notas, algún apunte adicional a las que se entreguen en esta evaluación, ni tampoco utilizará calculadoras o cualquier otro instrumento de cálculo automatizado. Desarrolle los temas de manera ordenada.  
Firme como constancia de haber leído lo anterior.

Firma

TODOS LOS TEMAS DEBEN SER ESCRITOS EN MATLAB

Tema 1 (20 puntos). La siguiente definición recursiva permite calcular la multiplicación de dos números enteros no negativos.

$$\text{multi}(p, q) = \begin{cases} 0, & q = 0 \\ p + \text{multi}(p, q - 1), & q > 0 \end{cases}$$

- Escriba una función recursiva  $\text{multi}(p, q)$ .
- Desarrolle una prueba de escritorio para  $\text{multi}(3, 5)$

Rúbrica: literal a (15 puntos), literal b (5 puntos)

```
a. function r=multi(p,q)
    if q==0
        r=0;
    else
        r=p+multi(p,q-1);
    end
```

```
b. r=multi(3,5)
r=3+multi(3,4)
r=3+3+multi(3,3)
r=3+3+3+multi(3,2)
r=3+3+3+3+multi(3,1)
r=3+3+3+3+3+multi(3,0)
r=3+3+3+3+3+0
r=15
```

*blog.espol.edu.ec / educasamas/*