

**CC1001 - Computación I**  
**Computación Numérica con MATLAB**  
**Profesor: Nelson Baloian**

A continuación se encuentran las soluciones de la auxiliar de computación numérica, las líneas en cursiva significan comentarios.

1.

```
//function y=pol(a,x)  
static public double pol(double[]a,double x){  
    int n=a.length;  
    //exponentes=0:length(a)-1; %0 1 ... n-1  
    double[]exponentes = new double[n];  
    for(int i=0; i<n; ++i)  
        exponentes[i]=i;  
    //potencias=x.^exponentes; %x ^0, ..., x ^(n-1)  
    double[]potencias = new double[n];  
    for(int i=0; i<n; ++i)  
        potencias[i]=Math.pow(x,exponentes[i]);  
    //y=sum(a.*potencias);  
    double s=0;  
    for(int i=0; i<n; ++i)  
        s += a[i]*potencias[i];  
    return s;  
}
```

2.

```
%area(a,b,n): calcular area de f en [a,b] usando n ptos  
function r=area(a,b,n)  
x=linspace(a,b,n);  
y=f(x);  
z=y(2:end-1);  
ancho=(b-a)/n;  
r=ancho/2*(y(1)+y(end))+ancho*sum(z);
```

3.

```
%static public double raiz(double a,double b,double epsilon){  
function r=raiz(a,b,eps)  
    %double fa=f(a), fb=f(b), x;  
    fa=f(a); fb=f(b);  
    %while(true){  
    while 0==0  
        %double m=(fa-fb)/(a-b), c=fa-m*a;  
        m=(fa-fb)/(a-b); c=fa-m*a;
```

```

    %x=-c/m;
x=-c/m;
    %if(b-a<=epsilon) break;
if b-a<=epsilon r=x; return
    %double y=f(x);
y=f(x);
    %if(signo(y)==signo(fa)){a=x; fa=y;} else{ b=x; fb=y;}
if signo(y)==signo(a)a=x; fa=y; else b=x; fb=y;end
end %while

```

Propuesto. Escribir una versión recursiva. Fijarse en las diapositivas de las clases(búsqueda binaria).