

Manual Básico de programación en GAUSS

Eduardo Moreno A.

14 de agosto de 2000

1 Características Básicas

1.1 Forma de programar

El programa se basa en instrucciones que uno ingresa a través de la línea de comando, asignándole valores a las variables, por ejemplo, si deseamos que $A = 5$, basta con ingresar:

```
(gauss) A=5;
```

y después podemos verificar el valor de A ingresando:

```
(gauss) A  
5.000
```

Estas instrucciones se pueden ir ingresando en la pantalla o se puede crear un archivo con ellas y después ejecutarlo con el comando

```
(gauss) run nombreadarchivo
```

Además de esto, se cuenta con todas las operaciones entre variables (+, -, *, /) y comandos (max, min, mod, etc...).

Definición de variables Esto ya se ha hecho antes, se hace por medio de la instrucción:

```
(gauss) a=5
```

Esto define una variable a de valor 5, este valor se puede ver con el comando

```
(gauss) a
```

Definición de Matrices Esto se hace de la misma forma, pero con llaves({}) en los extremos, y separando cada elemento por un espacio, y las filas por una coma (.). Por ejemplo para definir la matriz

$$B = \begin{pmatrix} 3 & 4 & 5 \\ 6 & 7 & 8 \\ 1 & 2 & 3 \end{pmatrix}$$

se hace mediante el comando

```
(gauss) B={3 4 5, 6 7 8, 1 2 3}
```

y el resultado se puede observar de la misma forma anterior. Para referirse a un término de la matriz, por ejemplo el 1er elemento de B ,

```
(gauss) B[1,1]
3.0000
```

Operaciones basicas Los operadores basicos son $+$, $-$, $*$, $/$, y funcionan **como matrices**, es decir el comando $A*B$ multiplica la matriz A por la matriz B, por lo que si queremos una multiplicación término a término se hace mediante el comando $(.*)$

Por ejemplo, pruebe que pasa con

```
(gauss) B*B
```

y con

```
(gauss) B.*B
```

Otras Operaciones Además de lo anterior, se pueden concatenar (unir) dos matrices en forma horizontal o vertical para formar así otras matrices. Los comandos son $\sim$ y $|$ respectivamente.

```
(gauss) B~B
(gauss) B|B
```

Vectores y matrices especiales Existen comandos particulares para algunos vectores y matrices comunes, por ejemplo

```
(gauss) C=ones(4,1)
```

crea una matriz de puros 1 de tamaño 4×1 (ie. Un vector de 1). Y

```
(gauss) D=zeros(5,5)
```

crea una matriz de puros 0 de tamaño 5×5 . Y

```
(gauss) err=rndn(4,1)
```

entrega una variable *err* de tamaño 4×1 con valores aleatorios entre 0 y 1.

Valores propios el comando

```
(gauss) E=eig(B)
```

entrega en un vector *E* los valores propios de la matriz *B*. Entonces si quisieramos obtener el valor del mayor valor propio (en módulo) lo hacemos con el comando

```
(gauss) maxvp=maxc(abs(eig(B)))
```

el que deja este valor en la variable, *maxvp*, y de la misma forma el mínimo valor propio se tendrá por el comando

```
(gauss) minvp=minc(abs(eig(B)))
```

Solución de un sistema Si tenemos un sistema de la forma $Ax = b$, donde *A* es una matriz de $n \times n$ y *b* es un vector de $n \times 1$, este se resuelve mediante el comando

```
(gauss) x=b/A
```

dejando en *x* la solución del sistema.

1.2 Funciones

Uno puede definir funciones, para después poder evaluarlas. La forma de definir una función es la siguiente:

```
(gauss) fn f(x)=x^2;
```

De esta forma, uno define la función $f(x) = x^2$, de esta forma, si uno define un vector x , uno puede obtener un vector $y = f(x)$ mediante el comando

```
(gauss) y=f(x);
```

2 Control de flujos

2.1 Comando for

El comando **for** hace repetir toda una serie de instrucciones un número definido de veces, la forma de ejecutarlo es **for i(inicio,final,paso)**. Un ejemplo,

```
for i(0,10,1);  
    A=A*A;  
endfor;
```

Este programa multiplica A 11 veces, pues hace a i partir como 0 y lo incrementa en 1 hasta que llega a 10.

2.2 Procedimientos

Un procedimiento es una forma de ahorrarse hacer todo un algoritmo repetidas veces, este procedimiento recibe unos parámetros y entrega un resultado. La forma de hacer esto en GAUSS es mediante el comando **proc**. Y se ejecuta de la forma:

```
proc NOMBRE (parametro);  
    local var;    <esto usa una variable var en forma temporal>  
    <Aquí se pone el procedimiento que se desea hacer>  
    retp(resultado);  
endp;
```

(OBS: Es MUY importante que todas las líneas tengan al final un ;)
la forma:

```
proc suma(a,b);  
    local c;  
    c=a+b;  
    retp(c);  
endp;
```

Una vez cargado esto, basta ejecutar en GAUSS

```
(gauss) suma(3,4)  
7.000000
```

3 Graficos

3.1 Aspectos Básicos

Para graficar, se debe cargar primero una librería gráfica, con el comando

```
(gauss) library pgraph;
```

Luego, se tienen los datos del eje x en un vector y los datos de las abscisas en otra matriz del mismo largo. Luego se ejecuta el comando $xy(vector_1, vector_2)$.

3.2 Ventanas con Gráficos

GAUSS permite poner múltiples gráficos en una misma ventana, e incluso permite la superposición de estas para poder comparar distintos gráficos. Las instrucciones para hacer esto son las siguientes:

begwind Inicializa el proceso de ventanas.

endwind Termina el proceso de ventanas.

makewind Crea una nueva ventana, de un tamaño y posición dado.

setwind Elige un numero de ventana específico.

nextwind Elige la ventana siguiente.

El único comando complicado es **makewind**, su estructura es la siguiente:

makewind(ancho,alto,posx,poxy,transparencia) **ancho** ancho de la ventana
alto alto de la ventana
posx posición a lo ancho de la esquina inferior izquierda.
poxy posición a lo alto de la esquina inferior derecha.
transparencia esto tiene valor 0 o 1, 0 indica ventana normal, 1 indica ventana transparente.

Ejemplo: Supongamos que tenemos datos de la abscisa x en la variable xm y los datos de $f_1(x)$ y $f_2(x)$ en las variables $y1$ e $y2$. Ambos grafos con la misma escala.

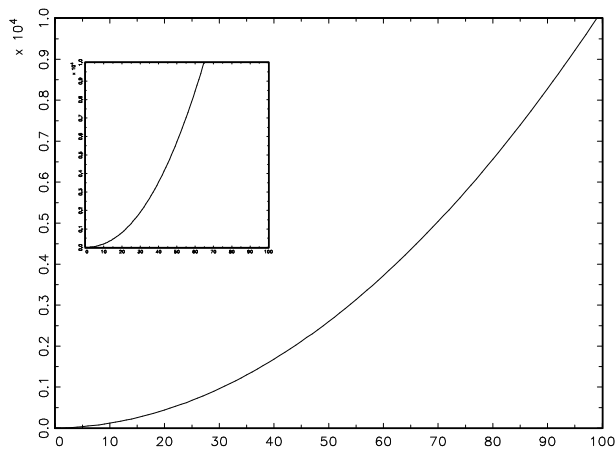
Queremos hacer un gráfico con 2 ventanas, en cada ventana una de las funciones. Para hacer esto basta hacer el siguiente programa:

```
library pgraph;
begwind;
makewind(10,10,0,0,0); /* Crea una ventana de 10x10 ubicada en (0,0)*/
makewind(3,3,1,3,0); /* Crea una ventana de 3x3 ubicada en (1,3)*/

ejex={0,100};          /* Fijo la escala del eje x */
ejey={0,10000};        /* Fijo la escala del eje y */
scale(ejex,ejey);

setwind(1);            /*elijo la primera ventana*/
  xy(xm,y1);
nextwind;              /*elijo la segunda ventana*/
  xy(xm,y2);
endwind;               /*Termino la contruccion de las ventanas */
```

Esto genera el siguiente gráfico:



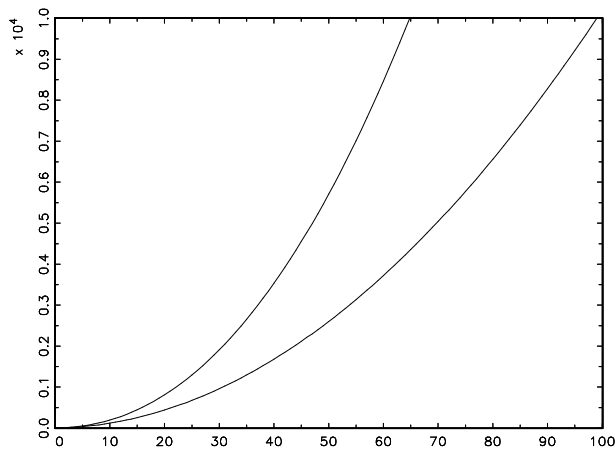
Ahora, si quisiéramos poner una ventana sobre la otra, habría que poner las ventanas del mismo tamaño, y elegir a la segunda ventana como “transparente”.

```
library pgraph;
begwind;
makewind(10,10,0,0,0); /* Crea una ventana de 10x10 ubicada en (0,0)*/
makewind(10,10,0,0,1); /* Crea una ventana de 3x3 ubicada en (1,3)*/

ejex={0,100};          /* Fijo la escala del eje x */
ejey={0,10000};        /* Fijo la escala del eje y */
scale(ejex,ejey);

setwind(1);            /*elijo la primera ventana*/
xy(xm,y1);
nextwind;              /*elijo la segunda ventana*/
xy(xm,y2);
endwind;               /*Termino la contruccion de las ventanas */
```

El gráfico ahora es el siguiente:



4 Gauss en el CEC

Para instalarlo en cualquier PC del CEC, uno debe hacer click con el botón DERECHO del mouse sobre el icono “Mi PC” y elegir *Map Network Drive*, elige un *Drive* (por ejemplo H:) y en la ventana de *Shared Directories* haz doble click en **ARAUCARIA** y luego en **GAUSS**. Con esto se creará una unidad H: en tu computador que dice “*Gauss en Araucaria*”.

Luego, elige del escritorio *Command Prompt*, se abrirá una ventana DOS, ahí cámbiate a Z: (escribiendo Z: y presionando ENTER) y escribe **H:GAUSS**. Con esto cargarás GAUSS en la pantalla después de unos segundos.

5 Documentos y Manuales

Están disponibles en la página web del curso los laboratorios efectuados a lo largo del año, y pueden usarse como ejemplos. Además, están en biblioteca los manuales de referencia y de utilización de GAUSS. Más información en la página web.