

PRACTICA I: GENERAR NÚMEROS

ALGORITMOS GENÉTICOS

3CM5

Colín Varela

Contents

1	Introducción	2
2	Desarrollo	2
2.1	Código	2
3	Pruebas	3
4	Conclusiones	4

1 Introducción

En esta practica veremos la habilidad de poder graficar números de una manera viable y directa para una fácil visualización de datos. Al alumno se le pide buscar la manera en la que pueda graficar usando cualquier librería en C o C++ que requiera.

2 Desarrollo

En esta práctica se opto por utilizar la librería 'plot-sdl' la cual toma provecho de las funciones de graficación de sdl para darle al programador una manera intuitiva y sencilla de poblar una gráfica con valores que posteriormente se puedan graficar.

2.1 Código

En general el código es bastante sencillo, iniciamos con las librerías, donde incluimos la librería de plot-sdl e igual incluimos las demás que usaremos para la generación de los números aleatorios

```
#include "stdio.h"
#include "plotsdl/plot.h"
#include "plotsdl/llist.h"
#include <time.h>
#include <stdlib.h>
```

La siguiente parte importante es donde generamos los valores aleatorios para graficar, aquí como se puede observar se toma un tope de 120 por practicidad.

```
//Generar numeros
int A[10];
int i;
srand(time(NULL));
for (i = 0; i < 10; ++i)
{
    A[i] = rand()%120;
}
```

Utilizamos la librería para agregar coordenadas basándonos en los valores generados. La sentencia: "push_back_coord(coordinate_list,0,0,A[0]);" Primero recibe un espacio de valores, luego el numero de linea, y por ultimo los valores x, y respectivamente

```
//push_back_coord
coordlist coordinate_list=NULL;
coordinate_list=push_back_coord(coordinate_list,0,0,A[0]);
coordinate_list=push_back_coord(coordinate_list,0,1,A[1]);
coordinate_list=push_back_coord(coordinate_list,0,2,A[2]);
coordinate_list=push_back_coord(coordinate_list,0,3,A[3]);
coordinate_list=push_back_coord(coordinate_list,0,4,A[4]);
coordinate_list=push_back_coord(coordinate_list,0,5,A[5]);
coordinate_list=push_back_coord(coordinate_list,0,6,A[6]);
coordinate_list=push_back_coord(coordinate_list,0,7,A[7]);
coordinate_list=push_back_coord(coordinate_list,0,8,A[8]);
coordinate_list=push_back_coord(coordinate_list,0,9,A[9]);
//print_list_coord(coordinate_list);
```

En esta parte le damos los últimos valores a la gráfica para que tenga un formato mas presentable.

```
//populate plot parameter object
plot_params params;

params.screen_width=800;
params.screen_height=640;
params.plot_window_title="Numeros Aleatorios";
params.font_text_path=argv[1];
params.font_text_size=18;
params.caption_text_x="";
params.caption_text_y="";
params.caption_list = caption_list;
params.coordinate_list = coordinate_list;
params.scale_x = 1;
params.scale_y = 10;
params.max_x = 9;
params.max_y = 120;
```

Finalmente graficamos y mandamos un éxito o un error dependiendo del desempeño al graficar.

```
if (ret == EXIT_FAILURE)
{
    printf("plot_graph return with status %d\n",ret);
    return EXIT_FAILURE;
}

return EXIT_SUCCESS;
```

Nota: Favor de leer el README.txt para mas información sobre como compilar y correr la practica

3 Pruebas

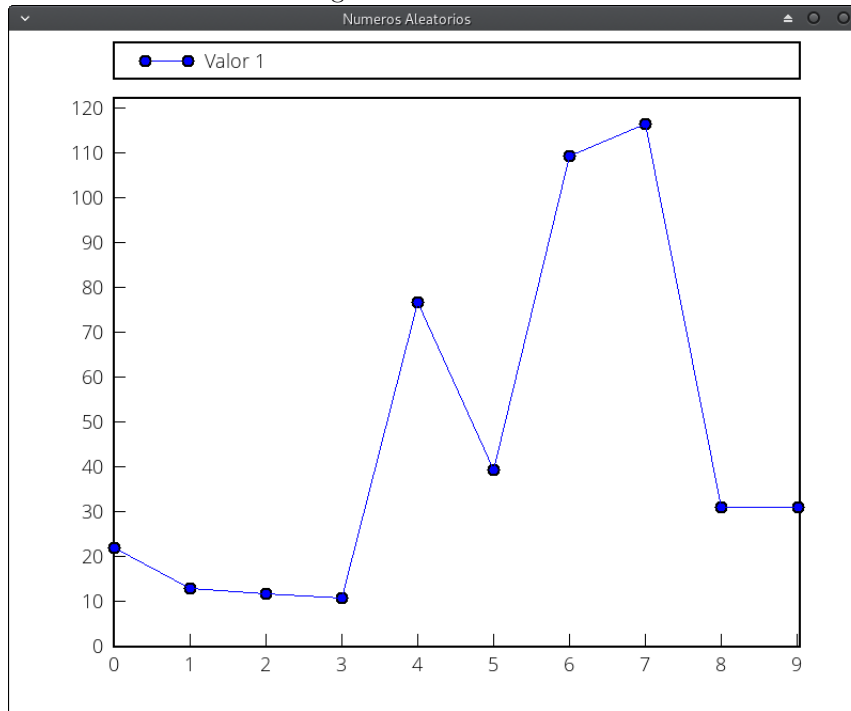
Primero abrimos una consola en la carpeta de la practica y compilamos con 'make', nos debe mostrar esta salida para una compilación exitosa.

```
Terminal - colin@Manjaro-de-Colin-42:~/Downloads/Practica 1 AG Colin_Varela
File Edit View Terminal Tabs Help
[colin@Manjaro-de-Colin-42 Practica 1 AG Colin_Varela]$ make
mkdir -p bin
mkdir -p release
mkdir -p release/include
cp -r plot-sdl-lib/plotsdl release/include/
ar rcs release/libplotsdl.a bin/plot.o bin/linked_list.o
gcc -Irelease/include -Lrelease plot-sdl-test/main.c -O3 -g -Werror -o release/p
lot -lplotsdl -lm -lSDL2 -lSDL2_ttf
[colin@Manjaro-de-Colin-42 Practica 1 AG Colin_Varela]$
```

Ahora vamos a correr la practica desde la misma carpeta con el siguiente comando:

```
Terminal - colin@Manjaro-de-Colin-42:~/Downloads/Practica 1 AG Colin_Varela
File Edit View Terminal Tabs Help
[colin@Manjaro-de-Colin-42 Practica 1 AG Colin_Varela]$ ./release/plot ./plot-sd
l-test/opensans.ttf
```

se puede observar que no genera ninguna salida en consola, pero nos abre una ventana nueva con lo siguiente:



Donde se aprecian los 10 números aleatorios tal como se requería

4 Conclusiones

La practica presiono a los alumnos a buscar una forma viable de graficar valores en código C , ahora que lo hemos conseguido se utilizara este conocimiento en practicas futuras para una fácil visualización de los datos procesados.