

66.20 Organización de Computadoras

Trabajo Práctico 0:

Infraestructura básica

Burdet Rodrigo, *Padrón Nro. 93440*
rodrigoburdet@gmail.com

Colangelo Federico, *Padrón Nro. 89869*
federico.colangelo@semperti.com

Manzano Matias, *Padrón Nro. 83425*
matsebman@gmail.com

2do. Cuatrimestre de 2014
66.20 Organización de Computadoras
Facultad de Ingeniería, Universidad de Buenos Aires

23 de septiembre de 2014

1. Objetivos

Familiarizarse con las herramientas de software que usaremos en los siguientes trabajos, implementando un programa (y su correspondiente documentación) que resuelva el problema piloto que presentaremos mas abajo.

2. Resumen

En el presente trabajo, se implementó un algoritmo que permite graficar los conjuntos de Mandelbrot dados ciertos parámetros para permitir centrarnos en una región en particular de dicho conjunto. El programa fue realizado en c 99 en un entorno de desarrollo linux. Compilado en Linux y en un ambiente emulado NetBSD.

3. Desarrollo

3.1. Paso 1: Configuración de Entorno de Desarrollo

El primer paso fue configurar el entorno de desarrollo, de acuerdo a la guía facilitada por la cátedra. Trabajamos con distribuciones Linux y con el GxEmul proporcionado por la cátedra, emulando un sistema NetBSD.

3.2. Paso 2: Implementación del programa

El programa debe ejecutarse por línea de comando y la salida del mismo dependerá del valor de los argumentos con los que se lo haya invocado.

3.2.1. Ingreso de parámetros

El formato para invocar al programa es el siguiente:

```
./tp0 [OPTIONS]
```

Los parámetros válidos que puede recibir el programa son los siguientes:

-o,	-output	(Parámetro obligatorio. Especifica archivo de salida, - para stdout)
-r,	-resolution	(Resolución de la imagen de salida).
-c,	-center	(Centro de la imagen).
-w,	-width	(Ancho del rectángulo a dibujar).
-H,	-height	(Alto del rectángulo a dibujar).
-v,	-version	(Muestra la versión).
-h,	-help	(Muestra la ayuda).

3.2.2. Interpretación de parámetros

Para parsear los parámetros se usó la librería de GNU getopt, en particular se usó getopt_long para permitir el pasaje de parámetros largos.

4. Compilación del programa

Para poder compilar el proyecto, se debe abrir una terminal Linux dentro del directorio donde se encuentra el código fuente escrito en C, y ejecutar el siguiente comando:

```
gcc -Wall -std=c99 main.c -o tp01
```

Esto generara un archivo ejecutable, llamado *tp0*.

¹Requiere tener instalado el compilador *GCC*

5. Compilación del programa en NetBSD

Para poder compilar el proyecto en NetBSD, se debe ejecutar el comando:

```
gcc -Wall -std=c99 -S -O0 main.c
```

6. Corridas de prueba y Mediciones

En las figuras que siguen a continuación se muestran los comandos utilizados para ejecutar el programa y se puede apreciar los resultados de las diferentes pruebas que realizamos.

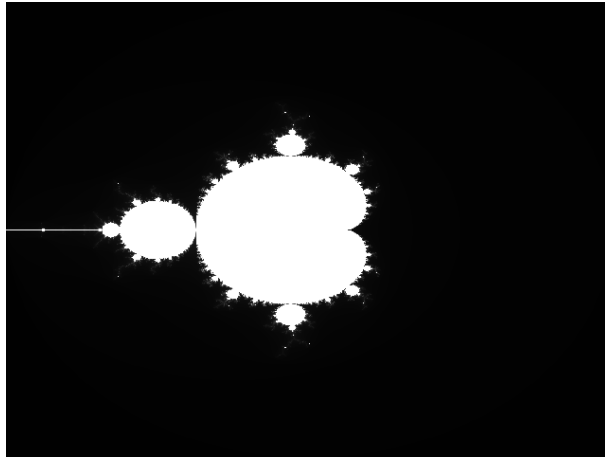


Figura 1: Llamada por defecto ./tp0 -o uno.pgm

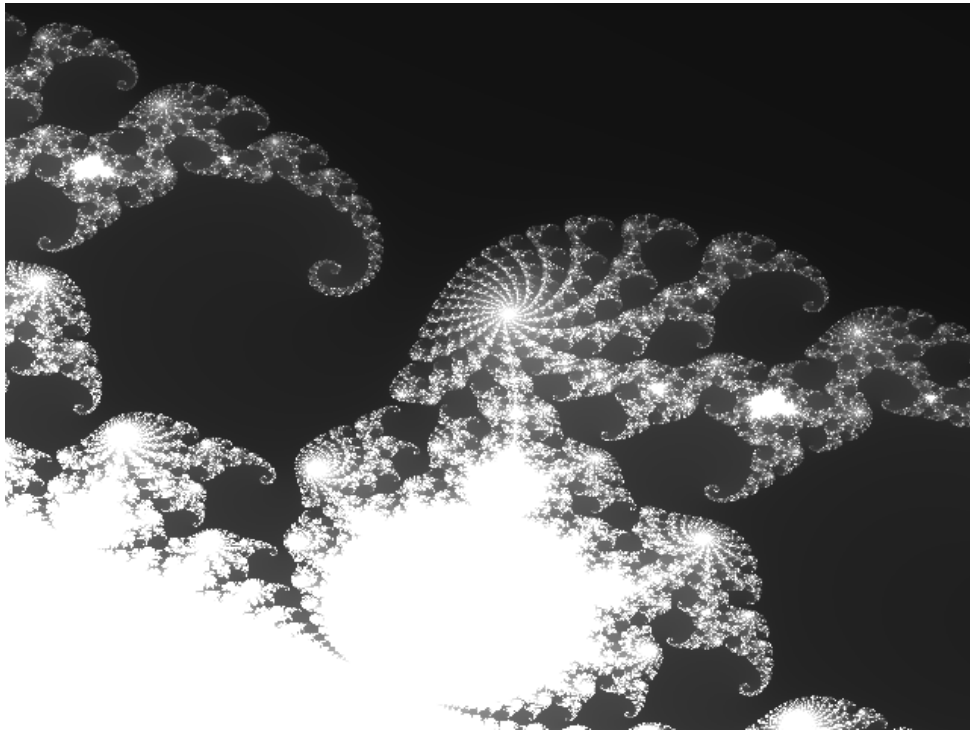


Figura 2: Llamada haciendo zoom sobre la región centrada en $(0.282, -0.01)$ con cuadrado de 0.005 de lado. `./tp0 -c +0.282-0.01i -w 0.005 -H 0.005 -o dos.pgm`

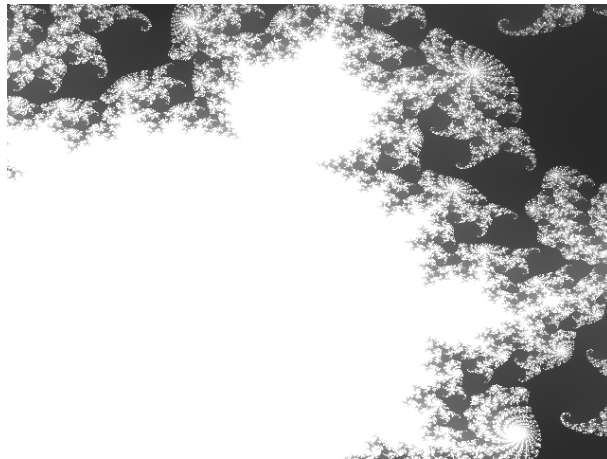


Figura 3: Llamada haciendo zoom sobre la región centrada en $(0.296, -0.02)$ con cuadrado de 0.003 de lado. `./tp0 -c +0.296-0.02i -w 0.003 -H 0.003 -o tres.pgm`

7. Conclusiones

Como se enuncia en el objetivo de este trabajo práctico, aprendimos a instalar y manejar el GxEmul, a realizar transferencias de archivos en Linux, así como también compilar y ejecutar programas en el NetBSD. Por otro lado, aprendimos a manejar y escribir informes en L^AT_EX. De este modo, estamos preparados para que en los próximos trabajos prácticos, nos aboquemos directamente al desarrollo de los mismos.