|  |
| --- |
| Laboratorio de Computación  Salas A y B |
| |  |  | | --- | --- | | Profesor: | Claudia Rodriguez Espino | | Asignatura: | Fundamentos de Programación | | Grupo: | 1104 | | No de Práctica(s): | 1 | | Integrante(s): | Valencia Mancera Erick Samuel | |  |  | |  |  | |  |  | | No. de Equipo de cómputo empleado: | #52 | | Semestre: | 1ª | | Fecha de entrega: | 21 de septiembre del 2018 | | Observaciones: |  | |  |  | |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Práctica #6: Entorno de C (editores, compilación y ejecución)**

**Objetivo:** Conocer y usar los ambientes y herramientas para el desarrollo y ejecución de programas en Lenguaje C, como editores y compiladores en diversos sistemas operativos.

**Desarrollo**

El lenguaje C es muy utilizado ya que la forma de dar instrucciones es muy cercana a lo que un humano podría abstraer, es decir, las instrucciones no son tal cual las que una computadora podría entender, para ello se necesitaría conocer a fondo el microprocesador, el sistema operativo entre otros aspectos. Por esta razón, C es conocido como un lenguaje de alto nivel, esto significa a que las instrucciones podrían ser entendidas fácilmente por un humano. En contraparte, un lenguaje de bajo nivel, son instrucciones que son cercanas a lo que la máquina puede entender y difícilmente pueden ser comprendidas por una persona que no tenga conocimientos de la máquina en que operarán.

Un programa en C se elabora describiendo cada una de las instrucciones de acuerdo a las reglas definidas en este lenguaje en un archivo de texto para después ser procesadas en un compilador. Un compilador es un programa que toma como entrada un archivo de texto y tiene como salida un programa ejecutable, éste tiene instrucciones que poden ser procesadas por el hardware de la computadora en conjunto con el sistema operativo que corre sobre ella.

**Editores de C**: Un programa en C debe ser escrito en un editor de texto para después generar un programa ejecutable en la computadora por medio de un compilador. Tanto el editor de texto como el compilador van de la mano con el sistema operativo y si posee o no interfaz gráfica por lo que son factores que se deben de tomar en cuenta a la hora de elegir el entorno para desarrollar programas en C.

**Editor Visual Interface de GNU/Linux (VI)**: El editor vi (visual interface) es el editor más común en cualquier distribución se sistemas operativos con núcleo basado en UNIX. Está disponible en línea de comandos y si el sistema operativo tiene entorno gráfico se puede acceder a él desde la terminal.

**GNU NANO**: Es un editor de texto disponible para sistemas operativos basados en UNIX en línea de comandos. Se puede acceder en un entorno gráfico desde la aplicación de terminal. Este editor es mucho más intuitivo que VI, aunque menos potente. No es necesario saber cómo se utiliza sino proporciona una interfaz que describe los comandos básicos. NANO es un editor clon de otro editor llamado PICO.

**GEDIT**: Es un editor de texto de entorno gráfico para sistemas basados en UNIX y se encuentra por defecto en el entorno de GNOME. Permite todas las funciones básicas que se pueden encontrar en la mayoría de los editores como copiar, pegar, guardar, etcétera, con las mismas combinaciones de teclas que ya son conocidas por los usuarios. También cuenta con menús en la parte superior que permite acceder a todas las funciones del editor por lo que no requiere conocimientos avanzados para poderlo usar.

**Compiladores**

Una vez codificado un programa en C en algún editor de texto, éste debe ser leído por un programa que produzca un archivo ejecutable. A este programa se le conoce como compilador y depende totalmente del hardware de la computadora y el sistema operativo que corre sobre ella.

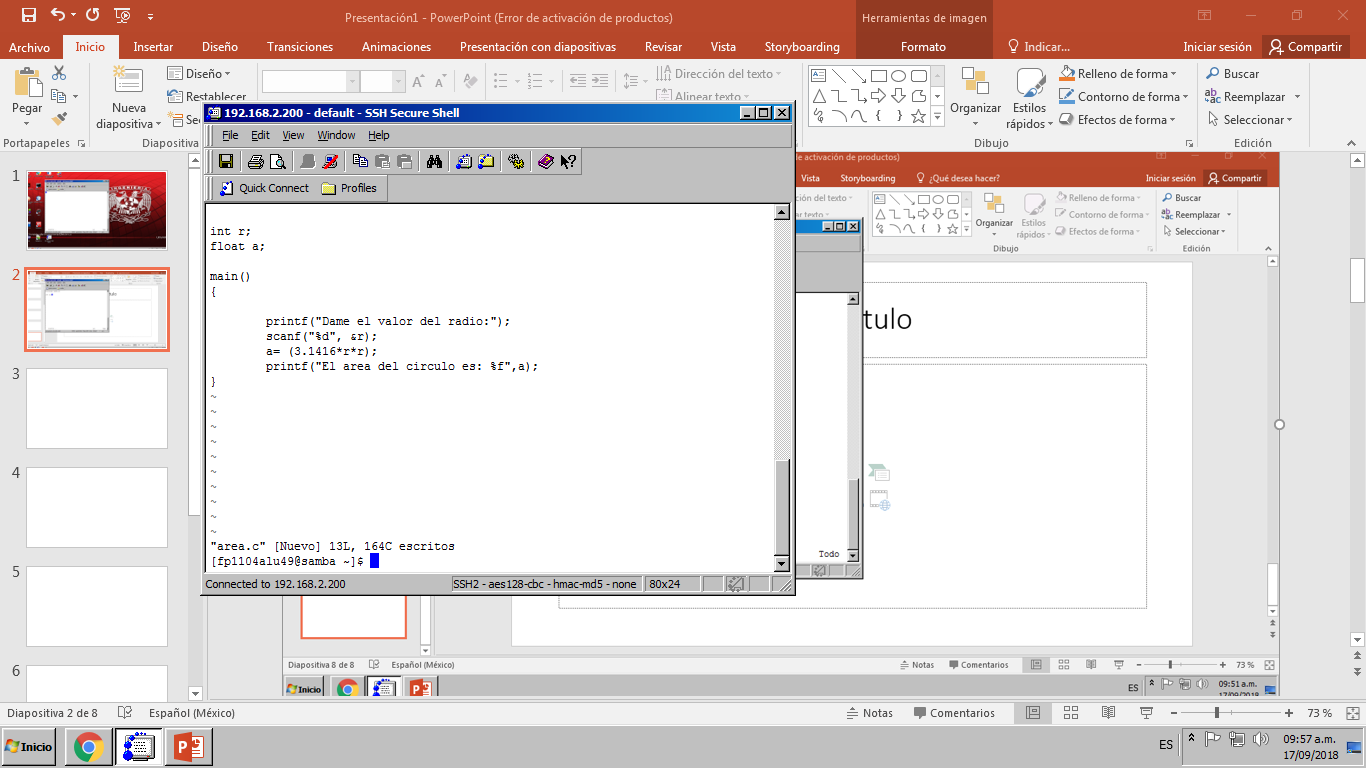
Recordando, un programa en C es universal, por lo que cada una de las instrucciones que lo conforman debe poder entenderlas muchos de los equipos en el mercado, aunque su naturaleza sea distinta.

**Actividades**

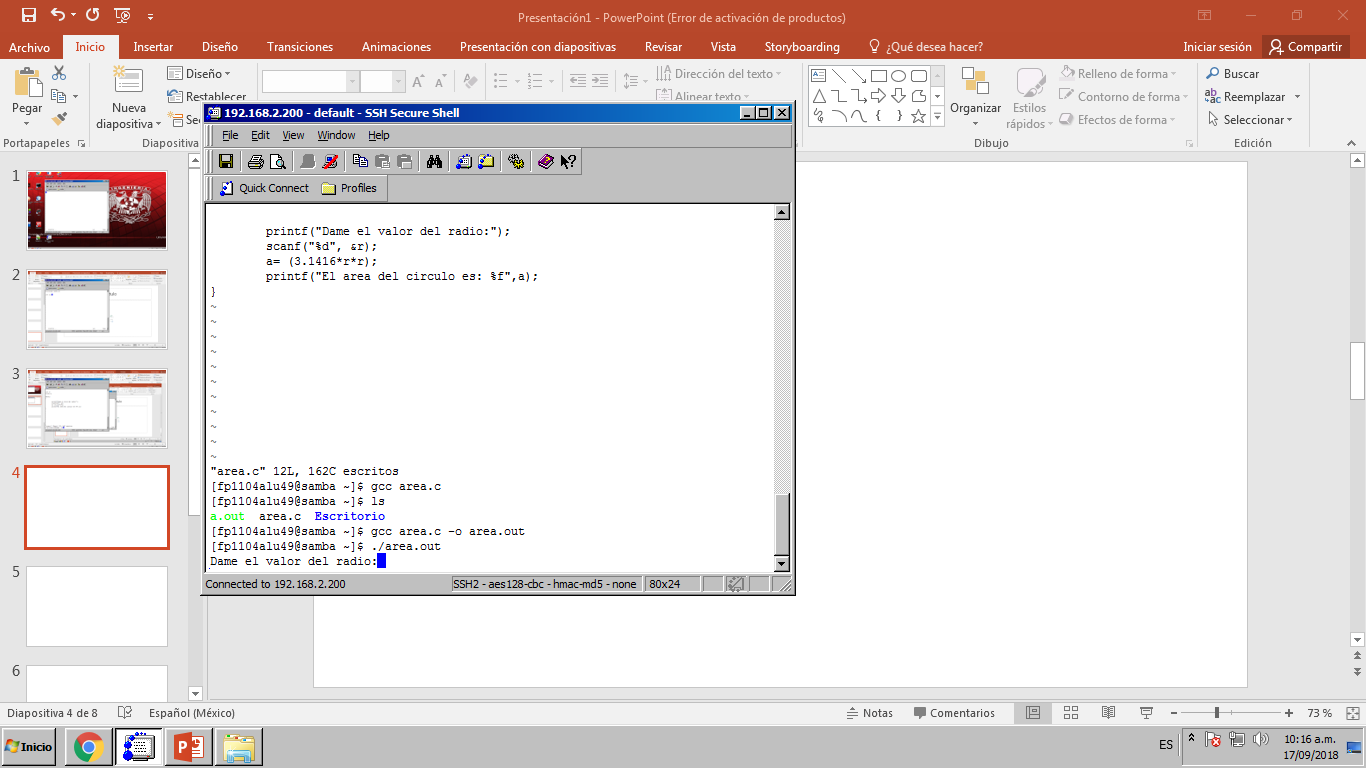
Abrimos el programa Secure Shell Client y utilizamos el comando **iv** para crear un programa.

1. Área de un Círculo

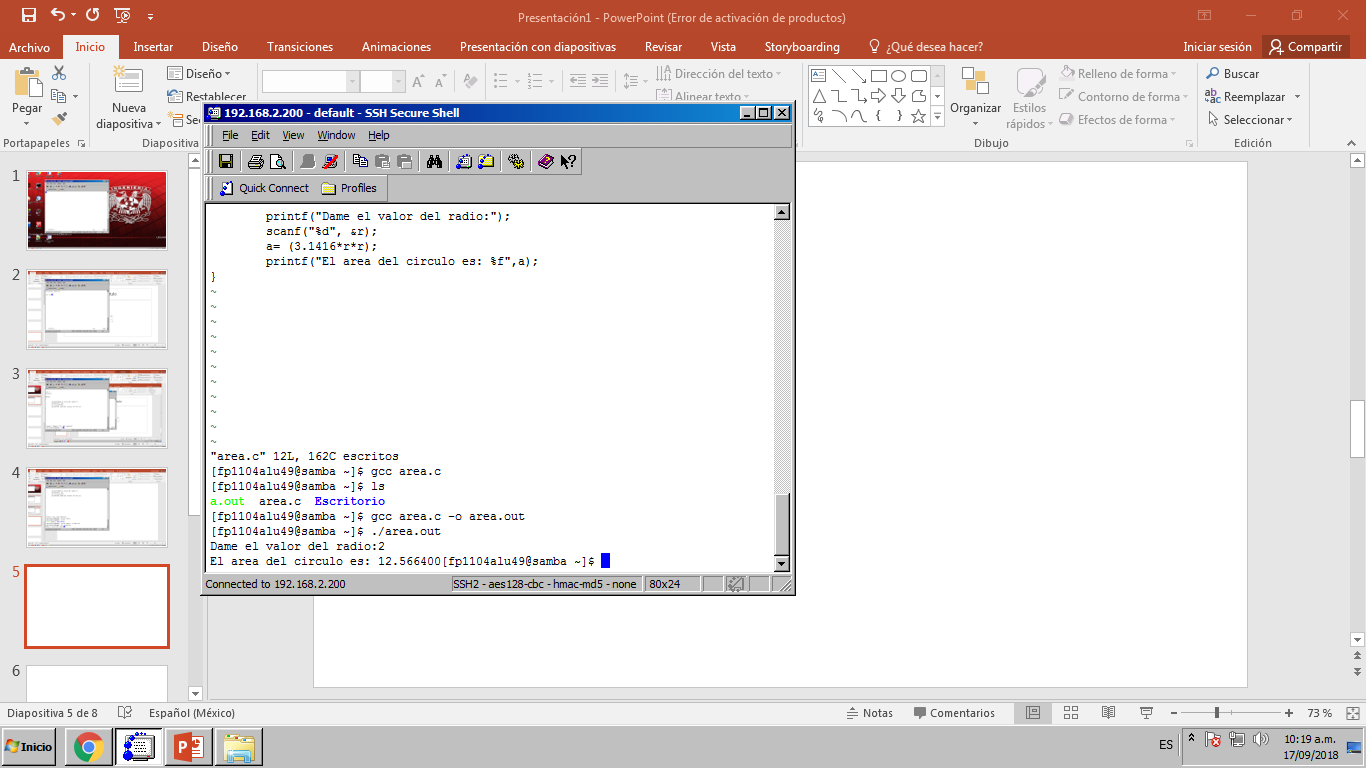
Utilizamos **iv** creando el programa *area.c*, con el cual escribimos el programa para encontrar el área de un circulo con un radio variable.



Utilizamos el comando *ls* para ver los archivos que ha en el escritorio, utilizando el comando *gcc nombredelarchivo* para compilar el programa. Al compilar el archivo encontramos un archivo llamado *a.out* y creamos otro utilizando *gcc archivo.c -o archivo.out*. En este caso, es *gcc area.c -o area.out*.

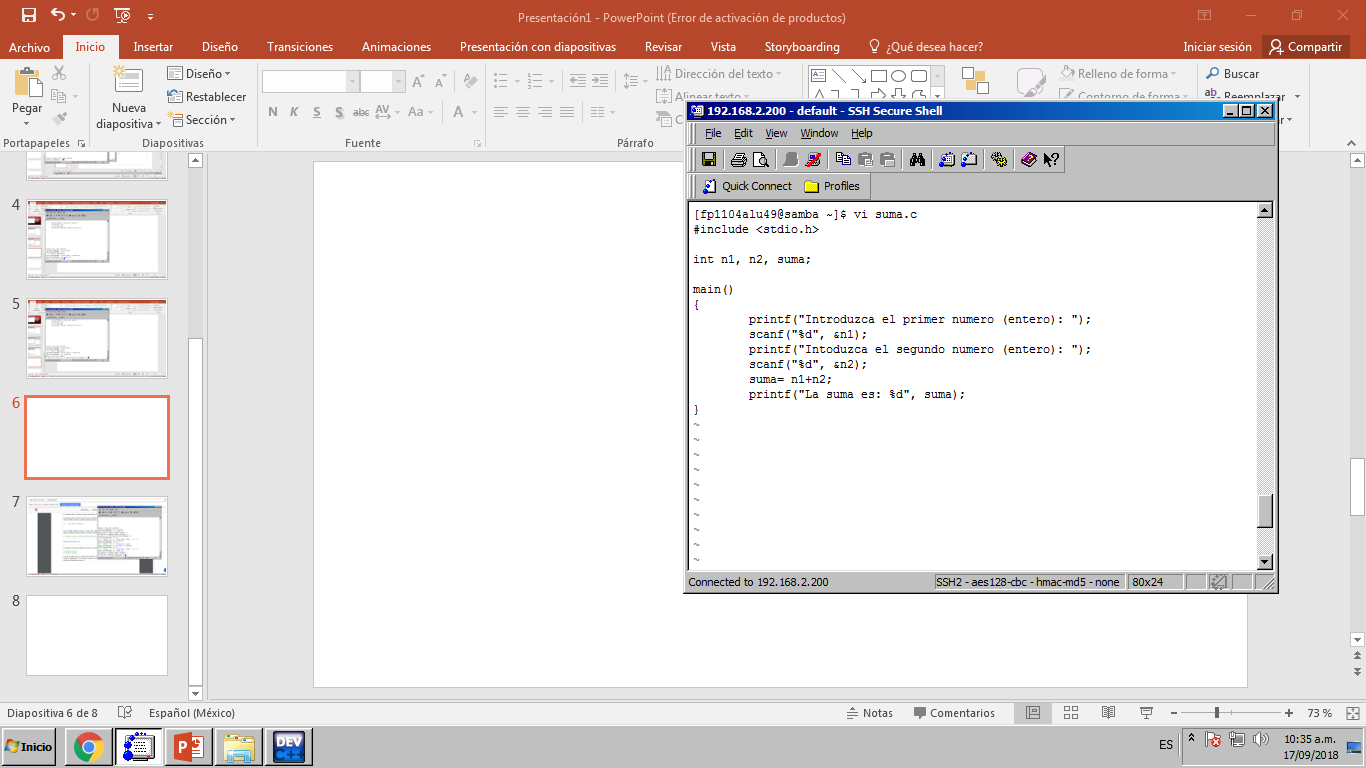


Y utilizando el comando *./archivo.out* para ejecutar el programa escrito. Y aquí usamos ./area.out. Y verificamos que el programa corra de manera correcta.

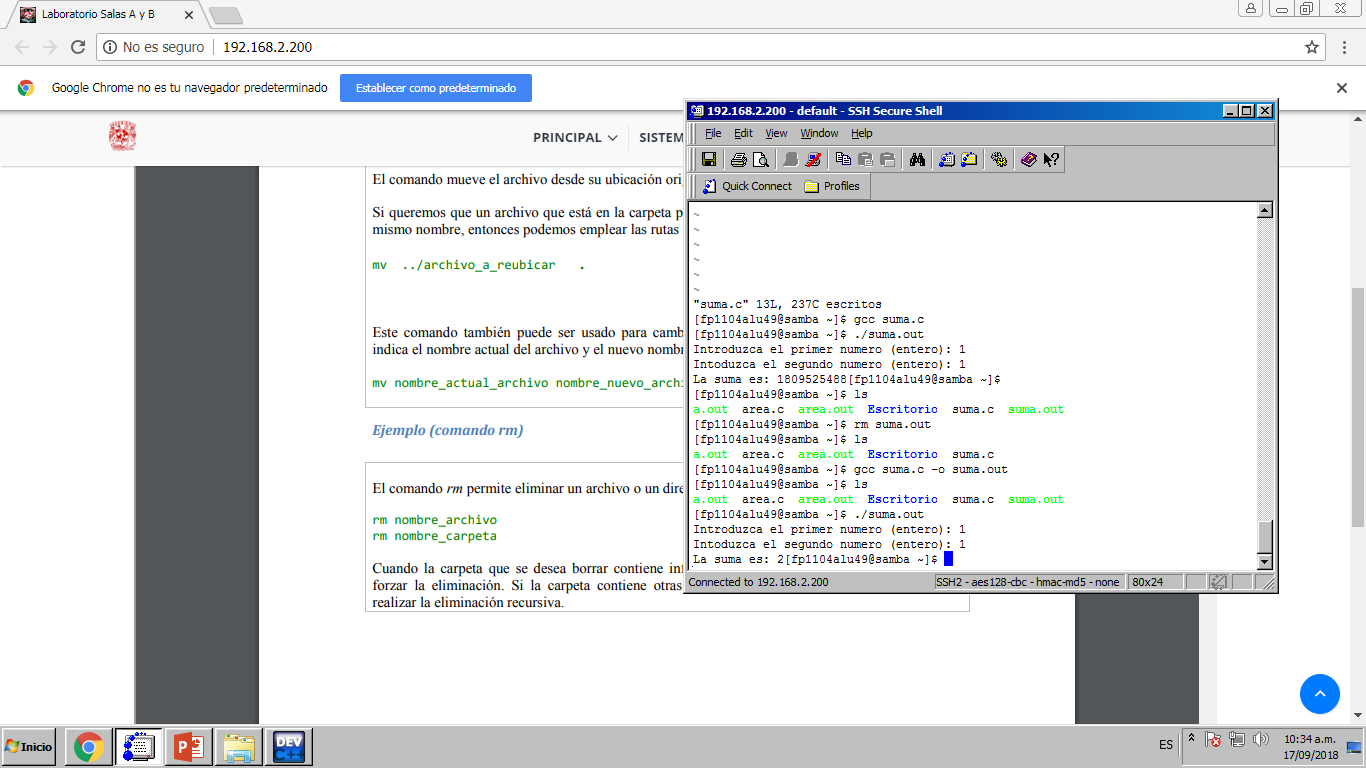


1. **Suma de dos números**

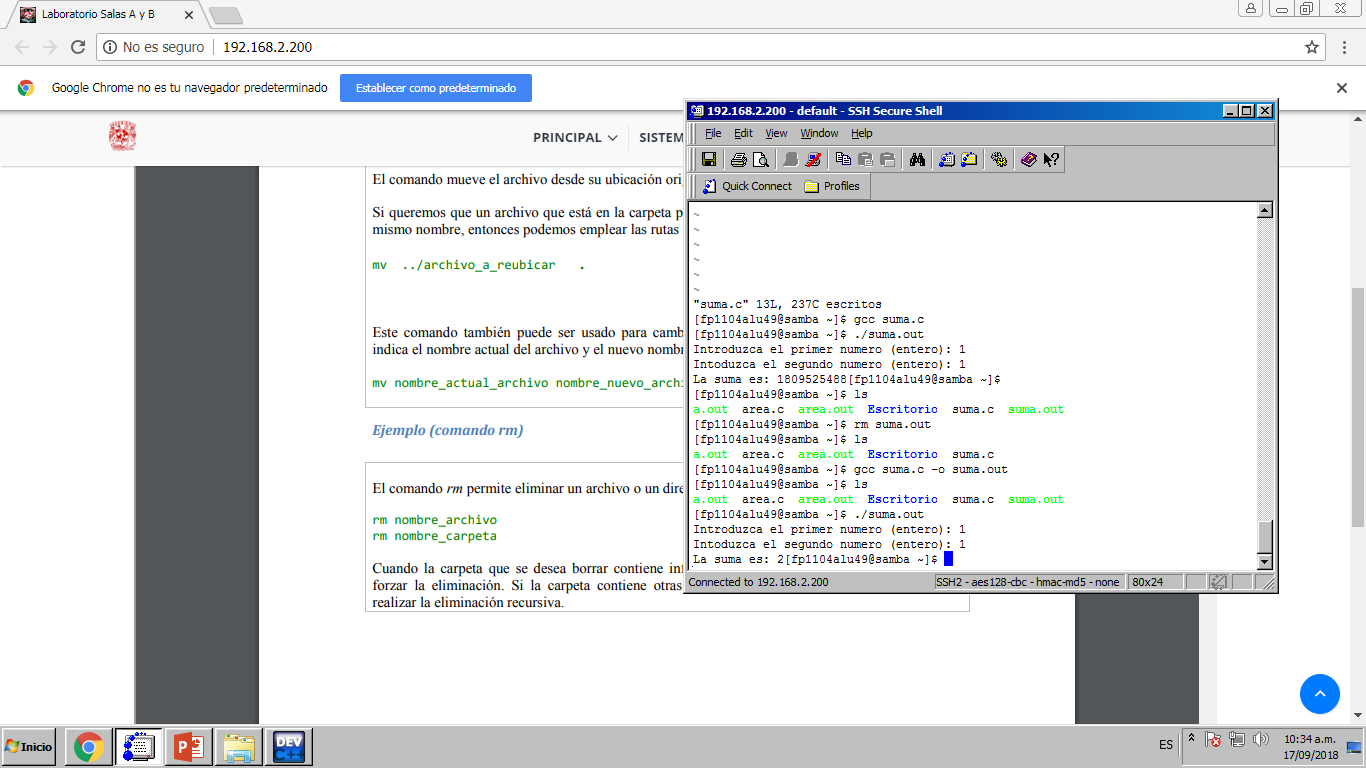
Creación de archivo **iv** *suma.c* y poner el progama.



Compilar el archivo *suma.c*, crear *suma.out*

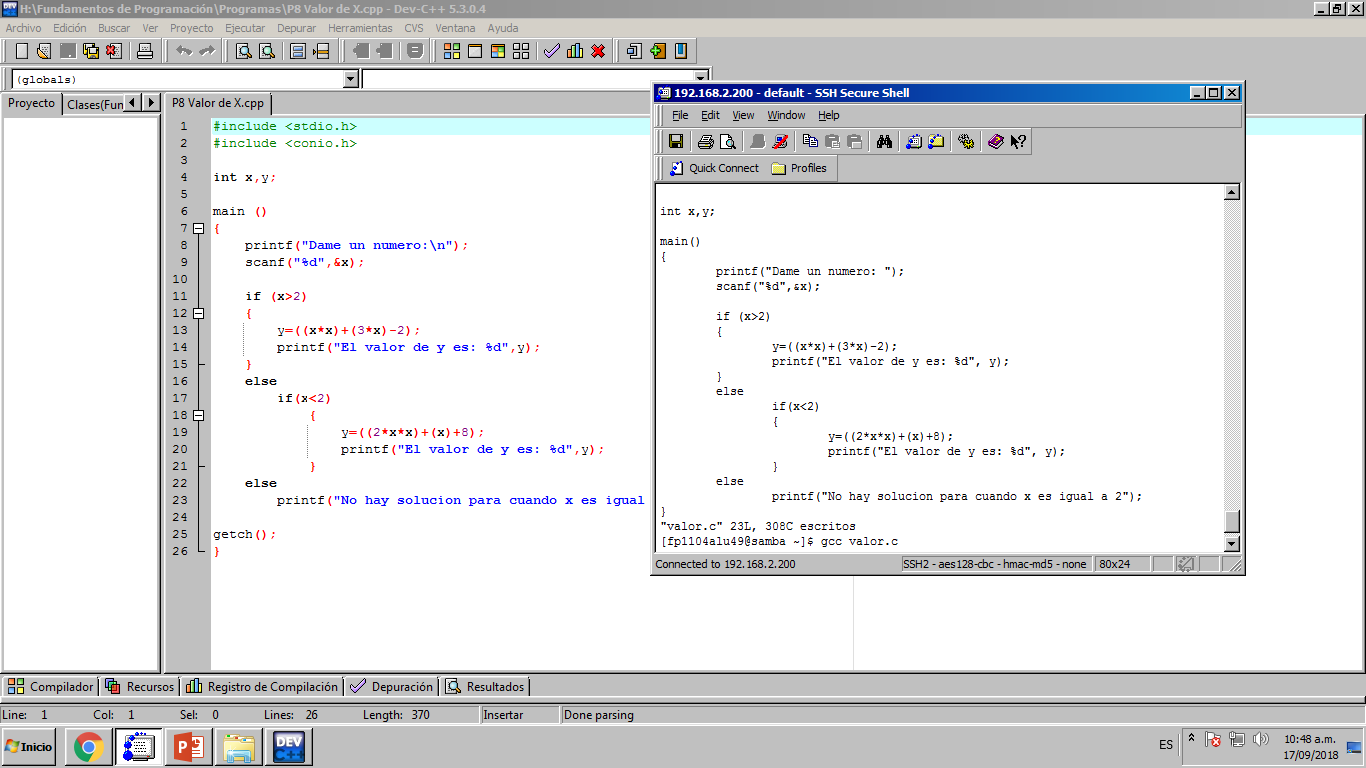


Ejecutamos el programa con *./suma.out* y verificamos que ejecute correctamente.

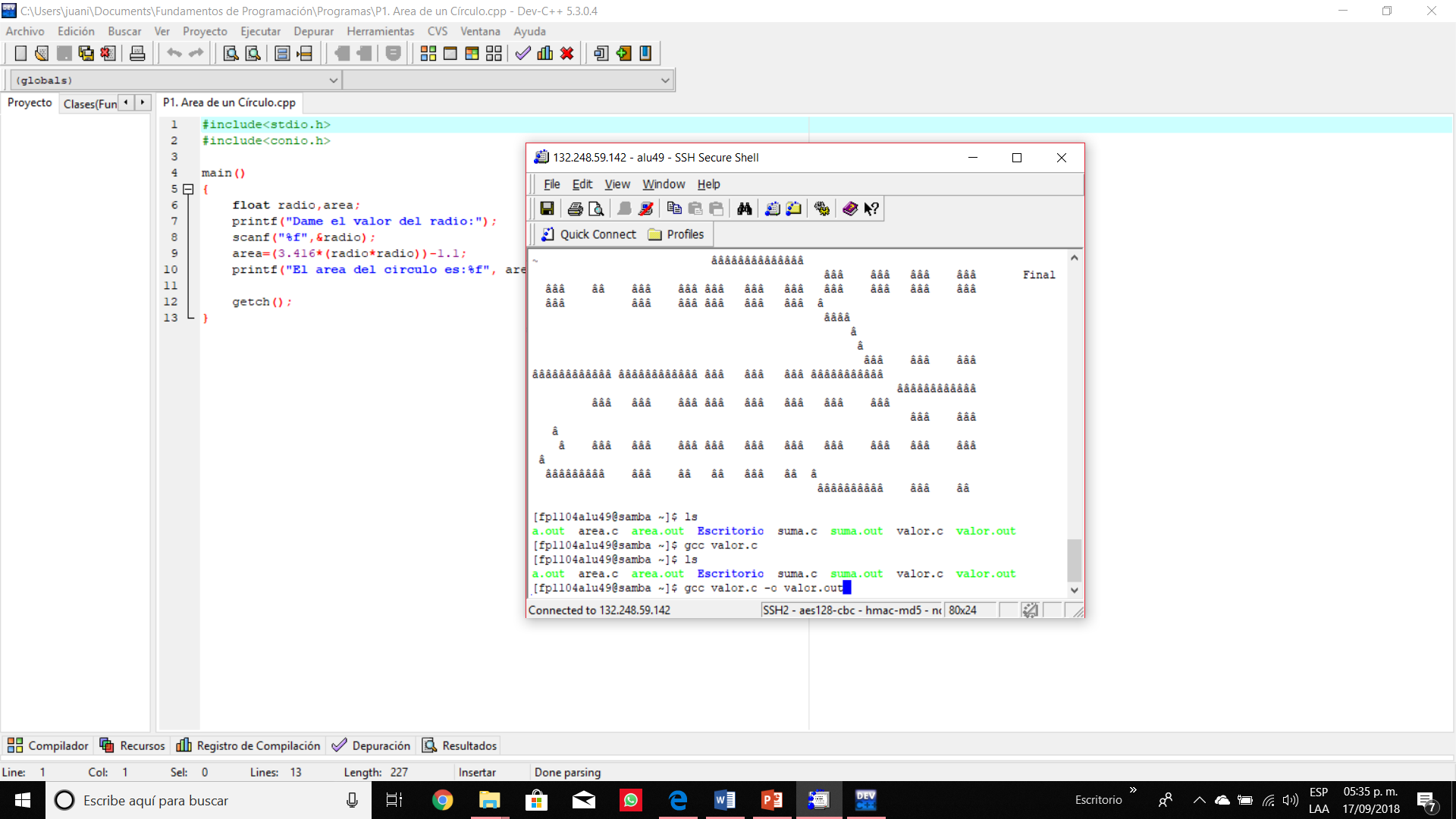


1. **Valores de X**

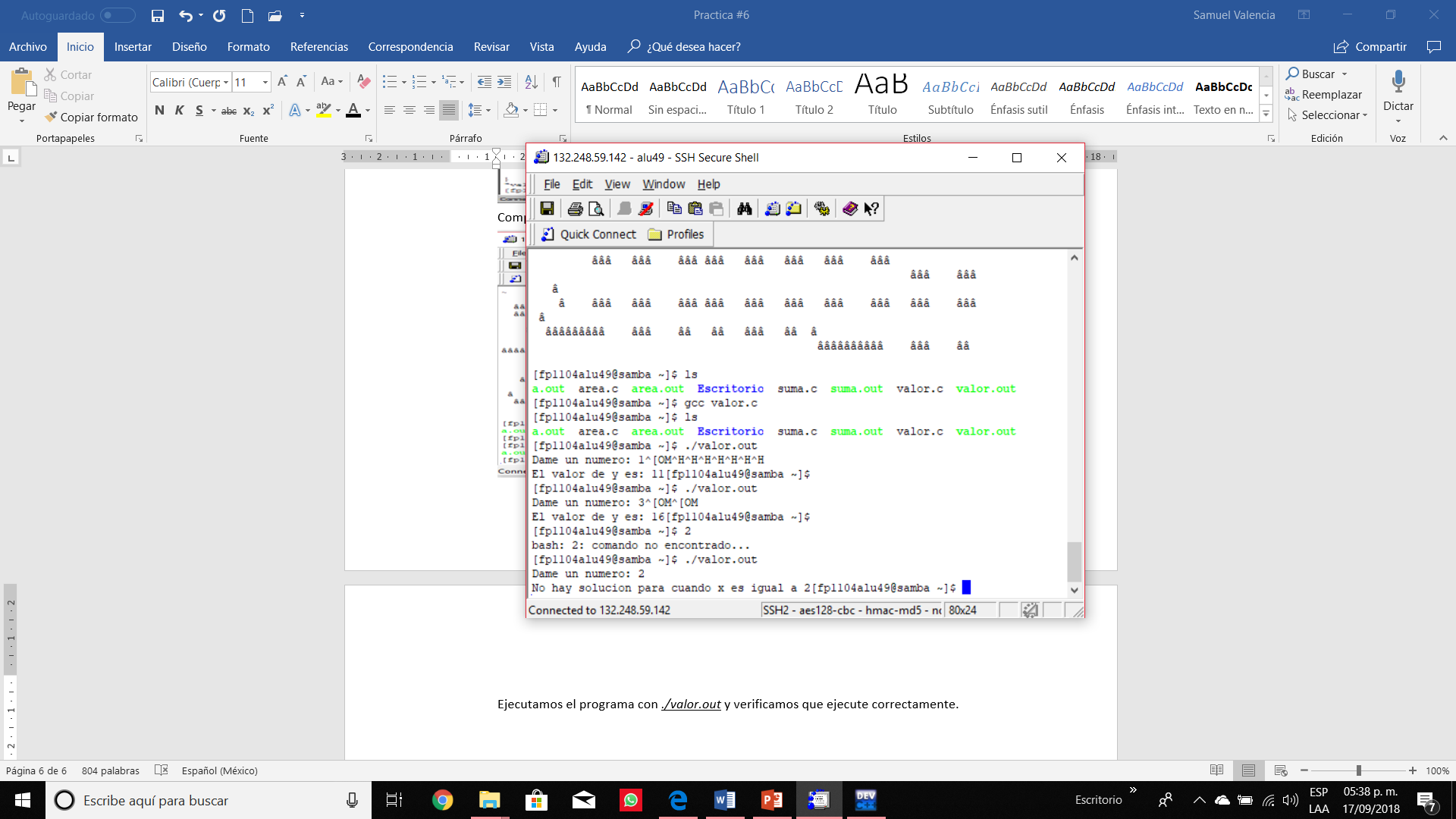
Creación del archivo **iv** *valor.c* y escribir el programa.



Compilar el archivo *valor.c* y crear *valor.out*



Ejecutamos el programa con *./valor.out* y verificamos que ejecute correctamente.



**Conclusiones**

Al ser ingenieros en computación, no podemos darnos el lujo de utilizar solo una forma de crear programas. Es importante saber utilizar los distintos formatos y programas que permiten crear códigos en distintos lenguajes.