

---

# 3

## "Hello, World!" en C

---

### 3.1. Objetivos

- Demostrar capacidad para editar, compilar, y ejecutar programas C mediante el desarrollo de un programa simple.
- Tener un primer contacto con las herramientas necesarias para abordar la resolución de los trabajos posteriores.

### 3.2. Temas

- Sistema de control de versiones.
- Lenguaje de programación C.
- Proceso de compilación.
- Pruebas.

### 3.3. Problema

Adquirir y preparar los recursos necesarios para resolver los trabajos del curso.

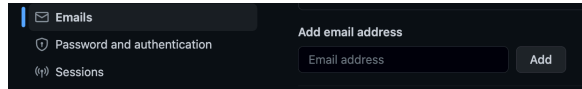
### 3.4. Restricciones

- Ninguna.

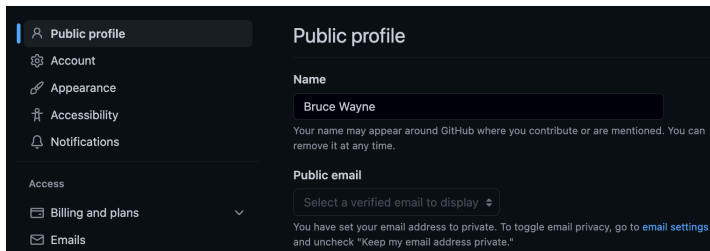
### 3.5. Tareas

1. Cuenta en *GitHub*
  - a. Si no tiene, cree una cuenta *GitHub*.

- b. Si no lo hizo, asocie a su cuenta *GitHub* el email `@frba` y verifíquelo. Es posible asociar más de una cuenta email a una cuenta *GitHub*.



- c. Si no lo hizo, indique que su cuenta email `@frba` es **pública**. Esto permite a la cátedra encontrar a los estudiantes. Si por temas de privacidad prefiere no tener como pública esa dirección, puede cambiarla al final del proceso.



## 2. Repositorio para público para la materia

- Cree un repositorio público llamado `ssl`.
- En la raíz de ese repositorio, escriba el archivo `readme.md` que actúa como *front page del repositorio* personal.
- Cree la carpeta `00-ChelloWorld`.
- En esa carpeta, escriba un segundo archivo `readme.md` que actúa como *front page de la resolución*.

## 3. Compilador

- Seleccione, instale, y configure, y pruebe un compilador **C11** ó **C18**. Los más osados pueden buscar un compilador que soporte **C2x**.
- Registre los resultados anteriores de la siguiente manera:
  - Indique en el `readme.md` el compilador seleccionado.
  - Pruebe el compilador con un programa `hello.c` que envíe a `stdout` la línea `hello, world!` o similar.
  - Ejecute el programa y verifique que la salida es la esperada.

- iv. Ejecute el programa con la salida *redireccionada* a un archivo `output.txt`; verifique su contenido.

#### 4. Publicación

- a. Publique el trabajo en el repositorio personal SSL la carpeta `00-ChelloWorld` con `readme.md`, `hello.c`, y `output.txt`.

#### 5. Armado de Equipo.

Aunque el trabajo es individual, fomentamos la colaboración entre compañeros para su resolución. Consideramos que es una buena oportunidad para armar equipo para los trabajos siguientes que en su mayoría son grupales. El docente del curso indica la cantidad de integrantes mínima y máxima por equipo.

- a. Informe el número de equipo en [esta lista](#)<sup>1</sup>.

Con el número de equipo y cuenta `@frba`, la Cátedra le envía la invitación al repositorio privado del equipo, por eso es importante que su cuenta *GitHub* tenga asociado como email público su email `@frba`, tal como indica el primer paso.

- b. Luego de aceptar la invitación al repositorio privado del equipo, si lo desea, puede cambiar el email público en *GitHub*.

### 3.6. Productos

```

Usuario                // Usuario GitHub
`-- SSL                // Repositorio público para la materia
  |-- readme.md        // Archivo front page del usuario
  `-- 00-ChelloWorld   // Carpeta el trabajo
    |-- readme.md      // Archivo front page del trabajo
    |-- hello.c        // Archivo fuente del programa
    `-- output.txt     // Archivo con la salida del programa

```

### 3.7. Referencia

- [\[Git101\]](#)

<sup>1</sup> [https://docs.google.com/spreadsheets/d/19MZodiTljD2WulmE8Y0WijNxIRdfL6vF\\_DvCn3uYIWg](https://docs.google.com/spreadsheets/d/19MZodiTljD2WulmE8Y0WijNxIRdfL6vF_DvCn3uYIWg)