

Trabajos de Sintaxis y Semántica de los Lenguajes

Esp. Ing. José María Sola, profesor.

Revision 0.1.0

Mar 2017

Table of Contents

1. Introducción	1
2. Normas Formales Generales para las Presentaciones de las Resoluciones	3
2.1. Repositorio del Equipo en GitHub	3
2.1.1. Contenidos del Repositorio	3
2.1.2. Front Page del Equipo	4
2.2. Carpetas para cada Resolución	4
2.2.1. Nombre de la Carpeta	4
2.2.2. Archivo <code>readme.md</code> de la Resolución	4
2.2.3. Header Comments (Comentarios Encabezado)	5
2.2.4. Contenidos Recomendados	5
3. @ Trabajo Práctico #0 — "Hello, World!" en C	7
3.1. Objetivos	7
3.2. Entrega	7
3.3. Secuencia de Tareas	7
4. Trabajo Práctico #1 — Conversor de Temperaturas	9
4.1. Objetivos	9
4.2. Restricciones	9
4.3. Entrega	10

1

Introducción

Los trabajos prefijados con (@) son comunes para todos los cursos.

Normas Formales Generales para las Presentaciones de las Resoluciones

Cada trabajo tiene sus requisitos particulares de presentación, esta sección indican los requisitos generales.

2.1. Repositorio del Equipo en GitHub

La presentación de cada resolución debe realizarse a través de *GitHub*, en una carpeta particular dentro del repositorio privado indicado en el curso para cada equipo.

2.1.1. *Contenidos del Repositorio*

El repositorio **sí** debe tener:

- un archivo `readme.md` que actúa como front page (carátula) del equipo,
- una carpeta por cada trabajo práctico resuelto, y
- un archivo `.gitignore`.

El repositorio **no** debe contener

- Archivos ejecutables.
- Archivos intermedios producto del proceso de compilación o similar.

2.1.2. Front Page del Equipo

Cada equipo debe diseñar una carátula del equipo, debe estar escrita con notación *Markdown* en el archivo `readme.md`, y debe tener, como mínimo, la siguiente información:

- Asignatura.
- Curso.
- Año de cursada, y cuatrimestre si corresponde.
- Número de equipo.
- Nombre del equipo (opcional).
- Integrantes del equipo actualizados, ya que, durante la transcurso de la cursada el equipo puede cambiar:
 - Usuario `github`.
 - Legajo.
 - Apellido.
 - Nombre.

2.2. Carpetas para cada Resolución

La resolución de cada trabajo práctico debe tener su propia carpeta en el repositorio del equipo.

2.2.1. Nombre de la Carpeta

El nombre de la carpeta debe seguir el siguiente patrón:

ÚltimosTresDígitosDelCurso-DosDelEquipo-DosDígitosNúmeroTrabajo-
NombreTrabajo

Por ejemplo, 051-02-00-Hello.

2.2.2. Archivo `readme.md` de la Resolución

Cada de resolución debe contar con un archivo `readme.md`, escrito en *Markdown* que contenga, como mínimo, la siguiente información.

- Número de equipo.
- Nombre del equipo (opcional).
- Autores:
 - Usuario github.
 - Legajo.
 - Apellido.
 - Nombre.
- Número y título del TP.
- Copia del enunciado.
- Hipótesis de trabajo que surgen luego de leer el enunciado.

2.2.3. Header Comments (Comentarios Encabezado)

Cada archivo fuente de debe comenzar con un comentario que indique el "Que", "Quienes", "Cuando" :

```
/* Qué: Nombre
 * Breve descripción
 * Quiénes: Autores
 * Cuando: Fecha de última modificación
 */
```

Example 2.1. Ejemplo de header comments

```
/* stack.h
 * Interface for a stack of ints
 * JMS
 * 20150920
 */
```

2.2.4. Contenidos Recomendados

- Archivo .gitignore.
- Archivo makefile.

- Tests.

3

@ Trabajo Práctico #0 — "Hello, World!" en C

Este es un trabajo individual.

3.1. Objetivos

- Demostrar con, un programa simple, que se está en capacidad de editar, compilar, y ejecutar un programa C.
- Contar con las herramientas necesarias para la resolución de los trabajos.

3.2. Entrega

- A las 13:00 del día de la segunda clase.

3.3. Secuencia de Tareas

1. Solicitar inscripción al Grupo Yahoo.
2. Registrarse en GitHub
3. Crear un repositorio público con el nombre `chellowor1d`.
4. Seleccionar, instalar, y configurar un compilador C11.
5. Probar el compilador con un programa `hello.c` que envíe a `stdout` la línea `hello, world!` o similar.
6. Ejecutar el programa, y capturar su salida en un archivo de texto `output.txt`.
7. Publicar `hello.c` y `output.txt` en GitHub.

8. Enviar a UTNFRBASSL@yahooogroups.com¹ usuario y repositorio GitHub.

¹ <mailto:UTNFRBASSL@yahooogroups.com>

4

Trabajo Práctico #1 — Conversor de Temperaturas

Este trabajo está basado en el ejercicio 1-15 de [K&R1988]:

1-15. Reescriba el programa de conversión de temperatura de la sección 1.2 para que use una función de conversión.

4.1. Objetivos

- Realizar el primer trabajo en equipo en el repositorio privado del equipo en **GitHub**.
- Demostrar conocimiento de:
 - Funciones.
 - Archivos header (.h).
 - Interfaces e Implementación.
 - Uso de make.

4.2. Restricciones

- Nombre del trabajo: conversor.
- Archivos:
 - `main.c`.

- `Conversor.h`.
- `Conversor.c`.
- `makefile`.
- Utilizar `const`.
- Utilizar `for` con declaración (C99).

4.3. Entrega

- A las 13:00 del día de la sexta clase, Abr 17.