

## Guía de Trabajos Prácticos

A continuación, se definen los trabajos prácticos entregables. Éstos son requisitos para regularizar la materia.

Todos los trabajos prácticos son grupales y se entregan digitalmente en horario de clases, ya sea en la computadora de laboratorio o en computadora personal o notebook del alumno.

NO IMPRIMIR MATERIAL: todas las entregas son en formato digital. De esta manera cuidamos el medio ambiente y evitamos gastos innecesarios.

### Trabajos Prácticos de la Guía de C++

- Aspectos a evaluar:
  - Diseño
  - Uso de git
  - Programación orientada a objetos en C++
- Entregables:
  - Diseño (diagrama de clases) en alguna herramienta
  - Código en un repositorio git
  - Programa funcionando
- Ejercicios:
  - Ejercicio 1: ensamblador/simulador
  - Ejercicio 2: estructuras de datos, uso de interfaces
  - Ejercicios 3a y 3b: gestor de Figuras.

### Trabajos Prácticos en C++ y un framework de desarrollo (Qt)

- Aspectos a evaluar:
  - Diseño
  - Uso de git
  - Programación orientada a objetos en C++, uso de Frameworks, aplicación de escritorio
- Entregables:
  - Diseño (diagrama de clases) en alguna herramienta
  - Código en un repositorio git
  - Programas funcionando
- Ejercicio: Gestor de sensores

### Al finalizar clases o en fecha de final: Trabajo Práctico integrador

- Aspectos a evaluar:
  - Diseño
  - Uso de git
  - Programación orientada a objetos en C++, uso de Frameworks, integración con HW (el HW puede ser simulado).
- Entregables:
  - Documento de Diseño en pdf
  - Manual del usuario en pdf
  - Código en un repositorio git
  - Programa funcionando
- El trabajo debe ser un prototipo funcional de tema libre, preferentemente relacionado con otras materias o trabajo final de la carrera.

### Clase especial de profundización temática

- Se debe presentar un área temática en profundidad. Las opciones son:
  - Virtualización y containers: máquinas virtuales (VmWare / VBox) vs containers (Docker, Kubernetes, LXC, etc)
  - Gestión de proyectos + Continuous Integration / Continuous Development / Dev Ops (ver Trello, Sinnaps, BitBucket)
  - Bases de datos: SQL Databases bases de datos relacionales (ie: SQLite, MariaDb, MySQL, PostgreSQL, Firebird, Oracle, Sql Server), JSon Databases (ie mongoDb, Elasticsearch) bases de datos orientadas a documentos, bases de datos de series de tiempo (ie: influxdDB, prometheus), bases de datos online (google Firebase / Firestore)
  - Buses de comunicaciones: zeroMQ, Rabbit, Qt DBus, etc
  - Cualquier otro tema propuesto por los alumnos y acordado con el profesor.
- Entregables:
  - Presentación (Power Point o similar)
  - Indicar referencias / fuentes