

Lineamientos generales para la elaboración, entrega y aprobación de los Trabajos Prácticos

- **Herramientas necesarias para resolver los ejercicios**
 - ✓ IDE C++, IDE Python, OOCASE para UML: instalados en la computadora personal del estudiante, usando la plataforma de su preferencia (Windows, Linux, MacOS, etc.)
 - ✓ Aplicativos o fragmentos de código para reutilizar (entregados por los docentes).
 - ✓ Hardware/Software adicional: tecnologías Arduino y/o Raspberry.
 - ✓ Discusión y ejemplos dados en clase.
 - ✓ Descripciones, ejemplos de código, URLs y referencias bibliográficas complementarias a las informadas en el programa de estudio, dadas en clase o dentro del material didáctico.
- **Metodología general**
 - ✓ Los temas fundamentales son tratados en clase y acompañados de apuntes.
 - ✓ La resolución de los ejercicios se realiza inicialmente en las horas de clase con el profesor y luego por cada alumno en horarios dispuesto por él en su ámbito personal de estudio.
 - ✓ Si bien no hay impedimentos en que los estudiantes se reúnan en grupos pequeños a analizar o discutir una solución, y/o eventualmente compartir un algoritmo específico, se aclara que el desarrollo de cada ejercicio de el/los TP/s de aprendizaje de esta guía debe realizarse de manera individual, y por lo tanto **cada estudiante debe presentar una solución lógicamente diferente** respecto al resto de sus compañeros.
 - ✓ En el caso del TP integrador, es viable trabajar en grupos reducidos de alumnos (máximo 3 integrantes) y presentarlo de manera conjunta. En este caso, cada alumno será evaluado y deberá explicar y defender su solución mediante coloquio individual, grupal o ante todo el curso.
 - ✓ En cada práctico se indican los elementos generales a usar, pero es el alumno quien debe investigar, ampliar, analizar y seleccionar según su criterio, los más apropiados en la resolución de cada ejercicio en particular.
 - ✓ Para la representación de los diagramas se usará una misma herramienta software, indicada en clase, para facilitar el intercambio.
 - ✓ Eventualmente, se proponen un conjunto de ejercicios y/o actividades complementarias a esta guía, indicadas en el aula cuya resolución es opcional. La finalidad de los ejercicios no obligatorios es la de mejorar las habilidades individuales de programación al mismo tiempo que proveen ciertas definiciones o implementaciones que puedan ser reutilizadas en la solución de los ejercicios obligatorios.
 - ✓ Es el estudiante quien debe lograr la habilidad de obtener software funcional en su computadora personal. Esto es así, tanto cada plataforma/IDE adoptado como para cada implementación realizada. El estudiante debe, con sentido crítico, verificar la correctitud de su solución a partir del producto final obtenido.
- **Forma y Fecha de Entrega**
 - ✓ Luego de resolver las consignas entregue un archivo comprimido (identificado con su apellido_legajo) que contenga:
 - o Código fuente de la solución a cada consigna obligatoria.
 - o Informe general de diseño, implementación y ejecución. Este informe debe realizarse conforme a las directivas dadas en cada caso.

- ✓ Un Trabajo Práctico inicia una vez habilitado en el aula virtual y dados los conceptos teórico-prácticos básicos correspondientes.
- ✓ Un TP puede estar asociado a una serie de entregas parciales.
- ✓ Se aconseja revisar las fechas de presentación de implementación e informe de trabajo práctico (indicadas en el aula virtual) de cada una de estas entregas. Esto permite, al docente realizar una revisión general y una eventual devolución con comentarios y discusión en clase, así como acciones de ajuste o corrección posterior por parte del estudiante. Una vez cerrada la fecha de entrega, sólo vuelve a habilitarse una fecha de entrega alternativa definitiva (una semana antes de finalizar el ciclo lectivo), no habiendo posibilidades de hacer ajustes o correcciones.

- **Aprobación**

- ✓ El trabajo práctico se considera aprobado cuando:
 - o Cada solución presenta un “desarrollo único” del problema, esto implica que no deben aparecer soluciones “gemelas” desde un punto de vista lógico (no sólo de aspecto).
 - o Se han aplicado los conceptos y se han incluido en la solución los elementos solicitados.
 - o La implementación de la solución y el informe, han sido entregados (en aula virtual) al profesor dentro de los plazos indicados oportunamente.
- ✓ En caso que el trabajo práctico no sea aprobado, en su desarrollo:
 - o El estudiante debe hacer cambios pertinentes según las observaciones realizadas por el docente y efectuar la entrega en la fecha alternativa correspondiente (si corresponde).
- ✓ Observaciones:
 - o Puede ocurrir que el trabajo práctico sea aprobado con observaciones. En ese caso, el docente indica, en coloquio, al estudiante cuáles son los aspectos observados y cuál habría sido el procedimiento correcto.
 - o Dado que, la verificación de los resultados obtenidos y de la lógica particular usada, es responsabilidad del estudiante, la evaluación de cada entrega, se realiza desde un enfoque documental, de aplicación de los principios de la OO y de lógica global.
 - o El formato y contenido del informe se indican/ejemplifican en clase.

- **Bibliografía**

- ✓ Apuntes de cátedra y referencias indicadas en clase
- ✓ DEITEL H. M. Y DEITEL P. J. (2009): Cómo Programar en C/C++ (6ª edición o anteriores), Prentice-Hall
- ✓ BRONSON, G. (2007): C++ para Ingeniería y Ciencias. 2da edición. Cengage Learning Editores S.A., México.
- ✓ STROUSTRUP, B. (2009): Programming. Principles and Practice Using C++, Pearson Education
- ✓ ACERA GARCIA, M.A. (2010): C/C++, Anaya Multimedia
- ✓ VON ROSSUM, G. (2017): El Tutorial de Python. Python Software Foundation 3ra edición.
- ✓ PEREZ CASTAÑO, A. (2016): Python Fácil. Marcombo
- ✓ BEAZLEY, D. y JONES, B. (2013): Python Cookbook. O'Reilly Media. 3ra edición