**Programación de Bajo Nivel**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | |  | | | | | |
| **Carrera** | | | | **Ingeniería Civil** | | | | | |
| **Código** | | | | **ICC–3102** | | | | | |
| **NRC** | | | | **4088** | | | | | |
| **Año de carrera/ Semestre** | | | | **3er año — 5o semestre** | | | | | |
| **Créditos SCT- Chile** | | | | **6** | | **Horas totales** | | | **180** |
| **Módulos semanales de docencia directa** | | | | **4** | | **Horas semanales de trabajo autónomo** | | | **6** |
| **Tipo de asignatura** | | | | **Obligatoria Licenciatura ICC**  **Parte de la concentración tecnológica “Diseño y Análisis de Algoritmos”** | | | | | |
| **Requisitos/ Aprendizajes previos** | | | | **ING–1103 Programación** | | | | | |
| **Profesor responsable/ Coordinador** | | | | **Jose Ossa** | | | | | |
| **Contacto** | | | | * **jossa1@miuandes.cl** * **Horario de atención de alumnos con inscripción a través de Canvas** | | | | | |
| **Definición de la asignatura** | | | | | | | | | |
| **Este curso enseña los fundamentos de programación en ambientes de bajo nivel, con lenguajes de programación compilados y gestión de memoria controlada por el programador. Se enfoca en el estudio de los lenguajes de programación C y C++, y entrega conocimientos prácticos sobre sistemas de compilación automatizada, directivas de preprocesamiento, enlanzamiento de programas con bibliotecas estáticas y compartidas, creación de bibliotecas, uso de depurador y perfilador. Con el lenguaje C se introducen conceptos de memoria de *stack* y memoria dinámica, uso de punteros a tipos y funciones, y referencias. Con el lenguaje C++ se introduce el manejo de memoria en el contexto de orientación a objetos, la abstracción de la entrada y la salida mediante flujos (*streams*), la sobrecarga de operadores, y conceptos como punteros a objetos y conversiones polimórficas, tablas de métodos virtuales, creación y uso de plantillas (*templates*), y uso de las plantillas de la biblioteca estándar de C++. Además, el curso enseña técnicas y herramientas para invocar código en C y C++ desde otros lenguajes interpretados o dinámicos como Python.** | | | | | | | | | |
| **Aporte al Perfil de Egreso** | | | | | | | | | |
| * **Posee habilidad para identificar y formular problemas abiertos de ingeniería, así como combinar la teoría y la práctica en su solución.** * **Tiene capacidad para seleccionar y utilizar las técnicas, habilidades y herramientas modernas de ingeniería adecuadas para la práctica de su profesión.** * **Posee las capacidades y la disposición para utilizar el pensamiento crítico en su quehacer profesional.** * **Es capaz de desarrollar software para la solución de problemas de distinta índole, utilizando varios lenguajes y plataformas.** | | | | | | | | | |
| **Resultados de Aprendizaje generales de la asignatura** | | | | | | | | | |
| * **Conoce los rudimentos del funcionamiento interno del computador y el manejo de memoria.** * **Tiene capacidad para seleccionar y utilizar el paso de parámetros por valor y referencia.** * **Es capaz de desarrollar software para la solución de problemas utilizando los lenguajes C y C++, y de integrarlo con otros lenguajes.** | | | | | | | | | |
| **Semanas** | | **Unidades** | | **Resultados de aprendizaje específicos de la Unidad** | | | | | |
| **2**  **3**  **2**  **1**  **1**  **1**  **3**  **3** | | **Introducción**  **Fundamentos de C**  **Memoria y punteros**  **Tipos de datos abstractos**  **Compilación avanzada**  **C++**  **Fundamentos de orientación a objetos**  **Librerías e interoperabi-**  **lidad** | | **Historia, arquitectura de von Neumann, binario, compilación**  **Control de flujo, tipos, operadores, ámbito (scope), streams**  **Stack y memoria dinámica, ciclo de vida, punteros a variables y funciones, paso por referencia, aritmética de punteros**  **struct, union, enum, typedef**  **comandos de preprocesador, makefile, dependencias, debugging.**  **Historia, streams, operadores especiales, strings**  **Clases, objetos, propiedades, métodos, constructores, new, delete, calificadores de acceso, namespaces, sobrecarga (de métodos, constructores y operadores), paso por referencia, herencia, templates**  **Librerías estáticas y dinámicas, wrappers para lenguajes interpretados** | | | | | |
| **Cronograma de actividades** | | | | | | | | | |
| **Se-ma-na** | **Fecha** | | **Contenidos/ Unidades Temáticas** | | **Estrategias o metodologías de enseñanza-aprendizaje** | | | **Metodologías de evaluación** | |
| **horas presen-ciales** | | **horas de trabajo autónomo de los estudiantes** |
| **1** | **4/8** | | **Introducción I** | | **2** | |  |  | |
| **2** | **11/8** | | **Introducción II** | | **4** | | **6** | **Laboratorio 0** | |
| **3** | **18/8** | | **Fundamentos de C I** | | **4** | | **6** | **Laboratorio 1A** | |
| **4** | **25/8** | | **Fundamentos de C II** | | **4** | | **6** | **Laboratorio 1B** | |
| **5** | **1/9** | | **Fundamentos de C III** | | **4** | | **6** | **Laboratorio 2B** | |
| **6** | **8/9** | | **Punteros I** | | **2** | | **8** | **Entrega T1** | |
| **7** | **15/9** | | **Punteros II** | | **4** | | **6** | **Laboratorio 3B** | |
| **8** | **22/9** | | **TDA** | | **4** | | **6** | **Laboratorio 2A** | |
| **9** | **29/9** | | **Compilación Av.** | |  | | **8** | **Entrega T2** | |
| **10** | **6/10** | | **C++** | | **4** | | **6** | **Laboratorio 3A** | |
| **11** | **13/10** | | **Or. a Objetos I** | | **2** | | **8** | **Laboratorio 4B** | |
| **12** | **20/10** | | **Or. a Objetos II** | | **4** | | **6** | **Laboratorio 4A** | |
| **13** | **27/10** | | **Or. a Objetos III** | | **2** | | **8** | **Entrega T3** | |
| **14** | **3/11** | | **Librerías I** | | **4** | | **6** | **Laboratorio 5A** | |
| **15** | **10/11** | | **Librerías II** | | **2** | | **8** | **Laboratorio 5B** | |
| **16** | **17/11** | | **Librerías III** | | **2** | | **8** | **Entrega T4** | |
| **Requisitos de aprobación** | | | | **T (promedio tareas) ≥ 4**  **E (examen) ≥ 3**  **P (promedio ponderado) ≥ 4** | | | | | |
| **Nota final de la asignatura** | | | | **, donde:** | | | | | |
| **Evaluaciones** | | | | | | | | | |
| **Tareas (T)** | | | | **Media aritmética de la nota de las 4 tareas** | | | | | |
| **Laboratorios (L)** | | | | **Media aritmética de la nota de los 7 mejores laboratorios (de 8 evaluados).** | | | | | |
| **Examen (E)** | | | | **Examen escrito** | | | | | |
| **Recursos de aprendizaje** | | | | | | | | | |
| * **Clases presenciales** * **Laboratorios presenciales** * **Canvas, incluyendo:**   + **Ejemplos de código usados en clases y otros complementarios**   + **Presentaciones de apoyo para algunos temas** * **Reuniones personales o grupales con el profesor, previa inscripción** * **Libros de consulta sugeridos:**   + **Kochan, S. G. (2014). Programming in C, 4a ed., Pearson Education**   + **Stroustrup, B. (2013). The C++ programming language, 4a ed. Pearson Education**   + **Stroustrup, B. (2014). Programming: principles and practice using C++, 2a ed. Pearson Education.** | | | | | | | | | |