

INGENIERÍA EN MECATRÓNICA									
COMPUTACIÓN III									
DEPARTAMENTO		Ciencias de la Computación, Informática y Sistemas de Representación							
PLAN DE ESTUDIOS		CARÁCTER				DICTADO			
2015		X	Obligatoria		Optativa		Anual	X	Cuatrimstral
AÑO	MÓDULO	RÉGIMEN				CUATRIMESTRE DE CURSADO			
3	5		Teórica	X	Teórica-Práctica	X	Primero		Segundo
CARGA HORARIA TOTAL			75		CANTIDAD DE SEMANAS			15	

DISTRIBUCIÓN DE LA CARGA HORARIA	Carga horaria
Bloques	Presencial
Ciencias Básicas de la Ingeniería	75
Tecnologías Básicas	0
Tecnologías Aplicadas	0
Ciencias y Tecnologías Complementarias	0
<b>TOTAL</b>	<b>75</b>

CARGA HORARIA DESTINADA A LAS ACTIVIDADES DE FORMACIÓN PRÁCTICA	Carga horaria
	Presencial
Instancias supervisadas de Formación Práctica	42,0
Proyecto Integrador	10,5
Práctica Profesional Supervisada	0
<b>TOTAL</b>	<b>52,5</b>

CARGA HORARIA SEMANAL	Presencial
Teoría	1,5
Formación Práctica	3,5
<b>TOTAL</b>	<b>5</b>

### 1. **Fundamentación.**

En la mecatrónica se necesita realizar programas que cumplan entre otros temas con:

- *Eficiencia y rendimiento*: cuando se programan sistemas embebidos estos suelen tener recursos limitados en cuanto a memoria y capacidad de procesamiento.
- *Tiempo real*: muchas veces se necesitan realizar tareas en tiempo real, lo que significa que se deben realizar tareas y responder a eventos en tiempo determinado.
- *Portabilidad*: Muchas veces se necesita trasladar el código a distintas plataformas y arquitecturas de hardware.
- *Control de hardware*: se necesita tener relación directa con el hardware.
- *Estabilidad y confiabilidad*: los programas deben tener estabilidad y confiabilidad ya que comúnmente están destinados a aplicaciones críticas.
- *Bajo consumo de energía*: hoy en día esto es muy importante más en los dispositivos que funcionan a baterías.
- *Seguridad*: algo cada vez más importante cuando se trata de dispositivos que se encuentran conectados a la red.

Estas necesidades hacen que a la hora de programar un ingeniero tenga que desenvolverse en distintos paradigmas de programación. *Computación III* es una materia que combina estos diferentes entornos y modelos de programación dándole al ingeniero mecatrónico herramientas que lo ayudarán a realizar simulaciones, analizar datos adquiridos, poder brindar una interfaz de usuario moderna y fácilmente manejable, entre otras cosas. *Computación III* resulta una herramienta indispensable para los futuros ingenieros mecatrónicos.

### 2. **Objetivos.**

Se espera que los estudiantes puedan:

- 1) Comprender la estructura y los aspectos característicos de la programación orientada a objetos para la correcta aplicación del mismo.
- 2) Entender las diferencias de programación para los distintos paradigmas.
- 3) Conocer diferentes entornos de desarrollo para que pueda desenvolverse en cualquiera de ellos.
- 4) Realizar interfaces gráficas adecuadas para el uso fácil de los operarios.

### 3. **Competencias y Descriptores.**

La asignatura asegura el desarrollo de los contenidos mínimos previstos en el Plan de Estudios, los descriptores de conocimiento y los ejes y enunciados multidimensionales y transversales previstos en la Resolución CD N° 525/23.

Dicha Resolución del Consejo Directivo dispone, entre otras cuestiones, la contribución





de cada asignatura a la matriz de tributación de los descriptores de conocimiento. Asimismo, establece el aporte mínimo que cada asignatura deberá realizar para el desarrollo de las competencias y/o ejes y enunciados multidimensionales y transversales previstos en la normativa ministerial correspondiente.

La contribución de esta asignatura a los contenidos mínimos y a los descriptores de conocimiento, puede evidenciarse en los contenidos analíticos detallados en el apartado específico de este Programa Analítico.

En cuanto a las competencias y/o ejes y enunciados multidimensionales y transversales, los mismos se desarrollan en la intensidad prevista en la Resolución CD N° 525/23, siendo abordados de manera integral, articulando la metodología empleada, los contenidos analíticos, las actividades de formación práctica, las formas de evaluación, junto a los demás aspectos intervinientes en el proceso de enseñanza y aprendizaje llevado a cabo por esta asignatura.

Por ello, el presente Programa Analítico se ajusta a la normativa dictada por la unidad académica para dar cumplimiento a la normativa nacional vigente vinculada a la carrera.

#### **4. Contenidos.**

##### **4.1. Contenidos Mínimos.**

Arquitecturas de programación: Tryton, C++ Builder. Programación de sistemas embebidos. Programación de dispositivos móviles: Android.

##### **4.2. Contenidos Analíticos.**

**TEMA I:** Arquitecturas de Programación: Descomposición modular. Cliente-Servidor. Dirigida a Eventos. Arquitectura de tres niveles.

**TEMA II:** Tryton, C++ Builder, Qt, MBED. Conceptos y Características.

**TEMA III:** POO con C++: Objeto. Clase. Método. Mensaje. Herencia. Conceptos POO en C++. Creación de clases. Constructores y destructores. Control de acceso a miembros. Modificadores de protección (private, public, protected). Creación de objetos. Herencia (Clases base y clases derivadas). Redefinición de miembros en clases derivadas. Polimorfismo. Funciones virtuales. Clases abstractas.

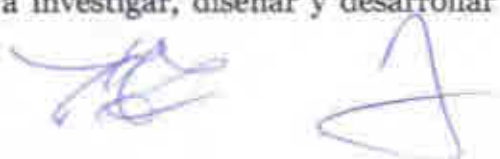
**TEMA IV:** Excepciones: Introducción a las excepciones. Sentencias try, throw y catch. Manejo de excepciones. Recuperación de errores usando excepciones.

**TEMA V:** Embebidos: Introducción. Conceptos. Programación.

**TEMA VI:** Dispositivos Móviles: Introducción. Conceptos. Entornos de programación. Android.

#### **5. Metodología de enseñanza y de aprendizaje.**

Se involucra al estudiante en la programación de los distintos paradigmas por medio de ejercicios y proyectos propuestos para cada unidad que provoquen en él, cuestiones que lo lleven a investigar, diseñar y desarrollar soluciones que le permitan adquirir técnicas de



resolución de problemas y fomentar el aprendizaje continuo.

Estos ejercicios o proyectos estarán debidamente organizados en forma tal que vayan poniendo a prueba todos los puntos necesarios para poder realizar los programas. Así, se les presentarán en clases todos los elementos, esto favorecerá el intercambio de las diferentes formas de realizar lo que uno desea con las herramientas del lenguaje, del paradigma de programación en particular, brinda para trabajar la resolución de un problema. Se seguirá el contenido de las unidades, las cuales contienen los distintos tópicos organizados en orden de los conocimientos necesarios.

En cada unidad, se proponen ejercicios de ejemplo que contemplan la mayoría de las posibilidades. La discusión del ejemplo pondrá en evidencia los problemas que pueda encontrar el alumno. Los ejercicios o proyectos propuesto en cada unidad alimentarán la generación de un proyecto final integrador orientado a la mecatrónica para que el alumno pueda trabajar en la identificación del problema y encontrar la resolución del mismo, esto hará que deba utilizar las herramientas tecnológicas adecuadas. Este proyecto deberá estar debidamente documentado ayudando a fomentar su capacidad de comunicación.

Se buscará que el estudiante tenga una participación activa de su aprendizaje, lo anime a experimentar y aplicar las soluciones, realizando debates en grupos y estudiando los distintos casos de resolución.

## **6. Descripción de las actividades Teóricas y de Formación Práctica.**

### **6.1. Actividades Teóricas.**

Las actividades teóricas se complementan inmediatamente con las actividades prácticas, se busca de esta forma que el alumno incorpore los tópicos dados en la teoría mediante trabajos prácticos, se entiende que para la programación esto es relevante ya que siempre se necesita poder visualizar el resultado de lo aprendido experimentado que el programa realizado hace lo esperado.

### **6.2. Actividades de Formación Práctica.**

Las actividades de formación prácticas se realizarán mediante trabajos prácticos que podrán enmarcarse como trabajos de laboratorio, resoluciones de problemas y ejercicios prácticos. Estos trabajos buscan que los alumnos afiancen los conceptos teóricos y prácticos mediante la realización de ejercicios que aborden los conceptos de Objeto, el manejo de Excepciones, y la creación de interfaces gráficas dinámicas. Se realizarán prácticas en donde utilicen placas de desarrollo de sistemas embebidos haciendo uso de las entradas y salidas digitales, timers y contadores que esta contiene. Por último, en un trabajo final los estudiantes van a integrar lo que han visto en los trabajos anteriores, agregando ahora la comunicación donde aplicarán protocolos de comunicación, la utilización de buffer circulares, entre otros, que le permitan controlar un sistema mecatrónico y visualizar e interpretar los datos enviados por el mismo.





### 6.3. ***Ámbitos donde se desarrollan las actividades de Formación Práctica.***

Para la realización de las clases teórica-prácticas se necesitará:

- Sala de computación
- Laboratorio de Mecatrónica

Respecto del aula esta debe ser amplia provista con computadoras con posibilidad de acceso a internet, capacidad para poder instalar todo el software que se utiliza en la programación de microcontroladores y el software para el desarrollo de las aplicaciones en PC. Las aulas deben constar con mesas largas para que se puedan apoyar todos los instrumentos y placas de desarrollo que se utilizarán en el transcurso del dictado de las clases. El aula debe tener una buena distribución de tomacorrientes, buena iluminación y acceso a los componentes e instrumentos básicos de electrónica.

### 7. **Articulación con otros espacios.**

*Computación III* es una materia que aporta verticalmente conocimientos esenciales para la mecatrónica sobre todo en temas de automatización y control con microcontroladores el cual es un dispositivo fundamental en cualquier sistema de control. Si bien es una materia de las ciencias básicas necesita de que los alumnos tengan una introducción a la arquitectura de las computadoras esto se articula bien con *Computación I* materia que trata este tema, y bien con *Informática y Sistemas de Representación* en donde se pone al alumno en la resolución de pseudocódigo. También el alumno debe tener conocimientos bien maduros sobre el lenguaje C, tema que se aborda en *Computación II*.

### 8. **Formas de evaluación.**

Se utilizará una evaluación formativa, para poder estar en todas y cada una de las etapas del proceso de aprendizaje. Así se podrá guiar a los estudiantes para que logren sus objetivos a medio y largo plazo. Se incitará a los alumnos a asumir un papel activo para que sean autosuficientes y, en el futuro puedan continuar con el proceso de aprendizaje con las habilidades y conocimientos adquiridos, cuando los estudiantes saben lo que están aprendiendo y cómo lo están estudiando, les resulta más fácil lograrlo por sí mismos. Para ello los docentes guiaremos y animaremos a los estudiantes a seguir por la buena dirección y mantener el ritmo.

Se utilizará una evaluación inicial para determinar el grado de conocimiento de los estudiantes, y adaptar así las estrategias de enseñanza, en pos de que los estudiantes puedan acomodar su enfoque de aprendizaje. El objetivo principal de estas evaluaciones para el aprendizaje es impulsar el deseo de los estudiantes de aprender, acentuando el desarrollo y el logro académico en lugar del fracaso.

Se emplearán para la evaluación proyectos, presentaciones y actividades grupales. En primera instancia, se evaluará al alumno por medio de ejercicios que indiquen el grado de



**RESOLUCION C.D. N° 667/23**

conocimiento de la materia, a partir de aquí se seleccionarán ejercicios niveladores en el caso que sea necesario. De esta manera la evaluación continua del alumno, por medio de ejercicios propuestos de forma que le permitan ver dónde necesita enfocarse. Por último, se realizará un trabajo final que involucre todos los temas dados para valorar todo aquello que el alumno ha aprendido y cómo logró integrar esos conocimientos en su día a día.

En todo momento se proporcionarán al estudiante detalles sobre cómo pueden mejorar su aprendizaje, se les brindará habilidades, ejemplos para que pueda resolver los problemas. Este hecho hace que muchos alumnos mejoren su rendimiento pero, sobre todo, que muchos otros no den por perdido el curso. También, se va a poner en claro en las clases que es lo que se va a enseñar, de esta forma el estudiante puede saber de antemano que va a aprender y pueda comprobarlo al final de las mismas. En todo momento se tratará de vincular lo que se está enseñando con su uso en la vida real así los estudiantes adquieren otra perspectiva; entienden mejor lo que están aprendiendo y para qué sirve.

## 9. Condiciones de Regularidad y Promoción.

### 9.1. Condiciones de Regularidad.

- **Resolución CD N° 200/12 - Artículo 14° - Inciso c):** Cátedras con un sistema de clases teóricas-prácticas 70% de asistencia para las clases dictadas.
- **Resolución CD N° 200/12 - Artículo 15° (los alumnos inscriptos en esta unidad académica, que presenten certificado de trabajo) - Inciso c):** Cátedras con un sistema de clases teóricas-prácticas 60% de asistencia para las clases dictadas.
- **Resolución CD N° 200/12 - Artículo 14° y 15° - Inciso d):** Haber presentado los informes de las actividades prácticas establecidos en la Programación de Cátedra.
- **Para esta asignatura esto último incluye:** La presentación que acredite la realización del 100% de los Trabajos Prácticos y Cuestionarios (Actividades de Formación Práctica Planificadas) será condición necesaria para regularizar y promocionar la asignatura.

### 9.2. Condiciones de Promoción.

- Tener aprobada la asignatura *Computación II*.
- La asignatura será promocionable con 2 (DOS) parciales, el último como proyecto integrador.
- Se deben cumplir los requisitos académicos para ser alumno regular del Reglamento Académico de la facultad (Resolución CD N° 200/12), de asistencia y actividades prácticas obligatorias.
- **Para esta asignatura:** La presentación que acredite la realización del 100% de los Trabajos Prácticos y Cuestionarios (Actividades de Formación Práctica Planificadas) será condición necesaria para regularizar y promocionar la asignatura.
- Obtener en ambos Parciales una calificación mayor o igual a 6 (SEIS).



**RESOLUCION C.D. N° 667/23**

- Existirá sólo una instancia recuperatoria: únicamente en los casos que, en sólo UNO de los parciales la/el estudiante haya obtenido una calificación menor o igual a 5 (CINCO) y mayor o igual a 4 (CUATRO), y estén cumplimentadas todas las otras exigencias para la promoción.
- En los casos de promoción, la nota final surgirá del promedio de las evaluaciones con calificación mayor o igual a 6 (SEIS).





## 10. Bibliografía.

Título	Autores	Editorial	Año de Edición
Fundamentos de Informática y Programación en C	Llanos Ferraris, Diego R.	Madrid: Paraninfo	2017
Mecatrónica: sistemas de control electrónico en la ingeniería mecánica y eléctrica. Un enfoque multidisciplinario (5a. ed.)	Bolton, W., Lomel Díaz, L. and Grillo Giannetto, M.	Alfaomega Grupo Editor	2013
Estructura lógica y diseño de programas	Cohen, Alan	Madrid Paraninfo	1986
Electrónica e informática aplicada	Rocha Díaz, Alberto Eduardo Ariel	Librería y editorial Alsina	2014

Company, T. (2023). Qt | Cross-platform software development for embedded & desktop. [online] Qt.io. Available at: <http://www.qt.io> [Accessed 16 Jun. 2023].

Arm Mbed. (2023). Home | Mbed. [online] Available at: <http://www.mbed.com> [Accessed 16 Jun. 2023].

St.com. (2023). Home - STMicroelectronics. [online] Available at: [https://www.st.com/content/st\\_com/en.html](https://www.st.com/content/st_com/en.html) [Accessed 29 March 2023].

