

# Programa analítico de Informática II - 2018

---

ASIGNATURA: **INFORMATICA II**  
DEPARTAMENTO: **ELECTRONICA**  
ÁREA: **TECNICAS DIGITALES**  
BLOQUE: **TECNOLOGIAS BASICAS**  
TIPO: **INTEGRADORA**

CODIGO : **95-0453**  
MODALIDAD: **Anual**  
Horas Sem : **5**  
Horas/año : **160**  
Nivel : **2**

---

## Objetivos generales:

---

- Adquirir sólidos conocimientos de programación para volcarlos a problemas de ingeniería, sobre la base de lenguajes estructurados modernos.
  - Promover el hábito por la correcta presentación de informes y desarrollar la habilidad para el manejo bibliográfico.
- 

## Objetivos particulares:

---

Que los alumnos:

- conozcan y dominen las estrategias de programación para interconectar procesos y dispositivos.
  - conozcan y dominen estructuras de datos avanzados de la programación.
  - conozcan los principios y herramientas de la programación orientada a objetos.
  - desarrollen habilidades para el manejo del hardware en bajo nivel mediante las técnicas de programación en "C".
  - conozcan y apliquen las técnicas de interrupciones del hardware para el desarrollo de programas en tiempo real.
- 

## Programa Sintético:

---

- 1.-** Programación avanzada en "C"
- 2.-** Listas y otras estructuras de datos
- 3.-** Aplicaciones de la Pc al cálculo numérico en temas de álgebra y análisis matemático
- 4.-** Filtros. Tratamiento de la información
- 5.-** Control de periféricos
- 6.-** Entornos gráficos
- 7.-** El lenguaje "C++"
- 8.-** Introducción a los Sistemas Operativos avanzados

---

## Programa Analítico:

---

### **UNIDAD TEMATICA 1: MULTIPROCESAMIENTO**

Threads(hilos de un proceso) y Forks.

IPCs (Comunicación entre procesos). Message Queues, Shared Memory, Semáforos. Dinámica de funcionamiento.

### **UNIDAD TEMATICA 2: EL LENGUAJE C++**

Presentación del lenguaje - C++ como evolución respecto del lenguaje C -

Tipos abstractos de datos: Clases - Clases y Objetos - Miembros públicos y privados - constructores y destructores - constructores parametrizados - Introducción a la programación orientada a Objetos - Concepto de encapsulamiento. Entrada/Salida en C++.

Operador visibilidad - Operadores para gestión dinámica de memoria: *new* y *delete* - Sobrecarga de funciones y operadores.

Funciones y clases *friend* - uso del apuntador *this* - calificador *const* - especificador de clase de almacenamiento: *extern* y *static*.

Herencia: tipos (*public*, *protected*, *private*) - clase base y derivada - miembro *protected* - constructores y destructores en la clase derivada.

### **UNIDAD TEMATICA 3: ENTORNOS GRÁFICOS**

Introducción a la programación en entornos gráficos - Librerías y entornos de desarrollo: Caso particular de QT y el QT Creator. - Introducción a la programación por eventos y las señales (Signals and Slots) de QT. - Formularios Básicos: Cuadros de Diálogo y aplicaciones de formularios sencillas. - Widgets mas comunes: Button, Check box, Radio button, Menu bar, Toolbar, Scrollbar, Text box, Combo box, Label y Otros. Ejemplos de aplicación: Programas Cliente y Servidor vinculados por IPC, acceso a Bases de datos, comunicación serie, Gráficos.

### **UNIDAD TEMATICA 4: Introducción a los microcontroladores y Sistemas Embebidos. Su programación en C.**

Concepto de Sistema Embebido. Presentación de periféricos elementales: Puertos de entrada/salida, Contador, Timer, UART, etc. Presentación del Kit de la Cátedra.

IDEs para microcontroladores - Compilación modular - Metodología de proyectos - Diagrama de capas: drivers, primitivas, aplicación - portabilidad. Concepto de Sistema Operativo en Tiempo Real.

ARCHIVOS DE CABECERA Y BIBLIOTECAS: Presentación de las funciones de biblioteca para uso del Kit. Ejemplos elementales.

PROGRAMACIÓN EN C: Tipos de datos - Análisis de variables asociadas - Configuración de registros - Diseño de funciones elementales. - Funciones de biblioteca - Ventajas y desventajas en la elección de utilizar funciones propias o funciones de biblioteca - Selección de la estructura de control de flujo apropiada para el diseño del programa. Uniones, operadores a nivel de bit

### **UNIDAD TEMATICA 5: PROGRAMACIÓN GOBERNADA POR EVENTOS**

DIAGRAMAS DE ACTIVIDAD: Utilización de las funciones de temporización de la Biblioteca del Kit de la Cátedra: *TimerStart* - *Timer\_Handler* - *SetTimer* - *GetTimer* - *StandByTimer* - *TimerStop* - *TimerClose*.

MAQUINA DE ESTADOS: Diagrama de globo. Implementación con *switch*. Implementación con múltiples *if*. Implementación con punteros a función.

Casos de análisis: Puerta corrediza, entrada garaje, escalera mecánica, sistemas de adquisición de datos, automatismos en general: llenadora, estampadora, etc.

Modelización del problema mediante la utilización de máquinas de estado - Casos de automatismos independientes dentro de un mismo equipo - Utilización de máquinas de estados independientes en paralelo: ventajas. Atención de múltiples temporizadores utilizando una máquina de estado.

### **UNIDAD TEMATICA 6: BIBLIOTECA ESTATICA - PRIMITIVAS**

Diagrama de capas: Las funciones primitivas de la Biblioteca utilizada con el Kit de la Cátedra por dentro: Escritura de Display LCD, Escritura de Display 7 segmentos. Lectura de buffer de teclado.

### **UNIDAD TEMATICA 7: BIBLIOTECA ESTATICA - FIRMWARE**

Diagrama de capas: Las funciones de firmware de la Biblioteca utilizada con el Kit de la Cátedra por dentro: Barrido de Display. Lectura de teclado matricial. Lectura de entradas digitales de microswitch: técnicas de absorción de transitorios (anti rebote). Escritura de salidas digitales.

### **UNIDAD TEMATICA 8: Introducción a la arquitectura de un Microcontrolador. Caso de uso: Cortex M3 y su programación en C**

Microcontrolador: Integración de periféricos en un solo chip. Concepto de Sistema Embebido. Familia Cortex M3.

Aplicaciones elementales de los puertos de entrada/salida de propósito general (GPIO): tratamiento como entrada, tratamiento como salida, tratamiento bidireccional. Memoria de datos. Memoria de programa - Mapa de memoria: RAM y ROM (flash). Especificadores de memoria.

### **UNIDAD TEMATICA 9: INTERRUPCIONES**

Idea general - Estrategias de atención - Fuentes interrupción externas - Funciones de interrupción - Pooling vs. Interrupciones - Registros asociados - Vector de interrupciones - Prioridades - Distribución de tiempos de ejecución para el programa principal y las funciones de interrupción - Mínimo tiempo admisible entre interrupciones - Pasaje de información entre el programa principal y las funciones de interrupción. Implementación en el Cortex M3.

### **UNIDAD TEMATICA 10: CONTADORES / TEMPORIZADORES**

Contador - Características principales - Temporizador: Caso particular de contador - Utilización como Contador (acumuladores de eventos) - Utilización como Temporizador (Base de Tiempo) - Multiplicadores de la base de Tiempo (ticks). Implementación en el Cortex M3 con systick - Registros asociados - Modos de funcionamiento. scheduler: Principios de funcionamiento. Real Time Clock (RTC).

### **UNIDAD TEMATICA 11: COMUNICACIÓN SERIE EN MICROCONTROLADORES.**

Necesidad de la comunicación serie - Serializadores y paralelizadores - Comunicación serie asincrónica: Conceptos.

UART Cortex M3 - Velocidad de transmisión - Registros asociados - Modos de operación - Buffers de Rx y Tx: Pilas y Colas - Implementación de protocolos punto a punto y multipunto - Estrategias de programación por pooling e interrupciones - Análisis comparativo.

### **UNIDAD TEMATICA 12: INTRODUCCIÓN A LA MEDICION Y GENERACION DE SEÑALES ANALOGICAS**

Introducción a la medición de magnitudes analógicas (temperatura, presión, humedad, etc.) - Conversores ADC: características principales - Estrategias de programación por pooling y por interrupciones - Interpretación de los valores

obtenidos desde el ADC (Tablas, filtros, etc.) - Eliminación de valores espúreos: Filtros de media móvil y de mediana. Implementación en Cortex M3 – Registros asociados. Conversores DAC – Características principales - Implementación en Cortex M3 – Registros asociados.

### **UNIDAD TEMATICA 13: APLICACIONES al ANALISIS NUMERICO**

Aplicaciones algorítmicas haciendo uso de métodos numéricos tradicionales. Soluciones aproximadas. Precisión. Errores.

#### **METODOLOGÍA DE LA ENSEÑANZA**

La aproximación a la enseñanza toma los aportes de los enfoques pedagógicos constructivistas y cognitivistas. Desde estas concepciones se trata de generar situaciones problemáticas para que los estudiantes pongan en juego sus ideas previas e interactúen con los nuevos saberes de forma tal que puedan construir nuevos conocimientos producto de esta interacción. Se trata de promover procesos cognitivos que permitan un abordaje integral para poder ir avanzando hacia la construcción de un proyecto integrador cuya presentación se realizará al final del curso. Se privilegia un enfoque grupal desde el punto de vista de la interacción de los estudiantes. Como otra estrategia metodológica se utiliza las presentaciones como disparador temático en soportes de diapositivas (ppt) para poner en disponibilidad ideas previas por parte de los estudiantes a modo fomentar aprendizajes significativos.

#### **BIBLIOGRAFÍA**

- **LPC17xx User manual** (2010), File name: UM10360 (bibliografía gratuita descargable del sitio [http://www.nxp.com/documents/user\\_manual/UM10360.pdf](http://www.nxp.com/documents/user_manual/UM10360.pdf)) [consultado en marzo de 2014].
- Deitel/Deitel(2004)–**Cómo programar en C/C++** -Prentice Hall–4ta.edición–Español
- Fatos Xhafa/Pere Pau Vazquez Alcocer/otros (2006) – **Programación en C++ para Ingenieros** – Thompson – 1ra. Edición
- Kernighan, B./Ritchie, D. (1991) – **El Lenguaje de Programación C** –Prentice Hall.
- Anónimo - **Aprenda Qt4 desde hoy mismo** (2010) (bibliografía gratuita descargable del sitio de la cátedra: [www.campusvirtual.frba.utn.edu.ar/especialidad/mod/resource/view.php?id=6671](http://www.campusvirtual.frba.utn.edu.ar/especialidad/mod/resource/view.php?id=6671))
- La Guía Beej de Comunicación entre procesos Unix <http://beej.us/guide/bgnet/> [En Inglés, consultado en febrero de 2018] o en [http://www2.electron.frba.utn.edu.ar/~mdoallo/descargas/guia\\_beej\\_ipcs\\_es.pdf](http://www2.electron.frba.utn.edu.ar/~mdoallo/descargas/guia_beej_ipcs_es.pdf) [En Español, consultado en febrero de 2018]

#### **BIBLIOGRAFÍA y RECURSOS WEB ADICIONALES RECOMENDADOS**

- Pont, Michael (2002) – **Embedded C** – Addison Wesley.
- Joseph Yiu (2009)-**The Definitive Guide to the ARM Cortex M3** – Newmes-2da Ed.
- Sitio Web oficial de Qt (en español): <http://wiki.qt.io/Main> [consultado en febrero de 2018]

- Recurso Web institucional de la cátedra:  
<http://www.campusvirtual.frba.utn.edu.ar/especialidad/course/view.php?id=13>
- Samek, Miro (2002) – **Practical Statecharts in C/C++** - CMP Books – Inglés

**PREREQUISITOS:**

	Cursadas:	Aprobadas:
Para cursar:	Álgebra y Geometría Analítica Análisis Matemático I Informática I	
Para rendir:		Álgebra y Geometría Analítica Análisis Matemático I Informática I