

**UNIVERSIDAD NACIONAL DE ASUNCIÓN**  
**FACULTAD POLITÉCNICA**  
**INGENIERÍA AERONAUTICA**  
**PLAN 2012**  
**PROGRAMA DE ESTUDIOS**

Resolución N° 17/10/05 – 00 Acta N° 998/08/05/17

### **I. - IDENTIFICACIÓN**

- |      |                                 |                |
|------|---------------------------------|----------------|
| 1.   | Asignatura                      | : Programación |
| 2.   | Nivel                           | : Tercero      |
| 3.   | Horas semanales                 | : 5 horas      |
| 3.1. | Clases teóricas                 | : 3 horas      |
| 3.2. | Clases prácticas                | : 2 horas      |
| 4.   | Total real de horas disponibles | : 80 horas     |
| 4.1. | Clases teóricas                 | : 48 horas     |
| 4.2. | Clases prácticas                | : 32horas      |

### **II. - JUSTIFICACIÓN**

Existen muchas razones del porque usar lenguaje C para Ingeniería; podemos citar entre ellas las que siguen a continuación:  
Los programas con C son casi tan rápidos como los lenguajes Asembler, lo cual es necesario para el control de sistemas eléctricos-electrónicos. Posee instrucciones que manipulan con facilidad bits, bytes y direcciones, lo cual es necesario para el control de dispositivos de E/S y para el gerenciamiento de la memoria.

Es un lenguaje estructurado, contamos con estructuras de control conocidas o bloques que permiten al programador un mejor control de las rutinas. Puede ser utilizado cuando queremos que un programa hecho en C en cualquier tipo de computadora, funcione en otra computadora sin problemas.

### **III. - OBJETIVO GENERAL**

Analizar las herramientas de programación del lenguaje C para la resolución de algoritmos.

### **IV. - OBJETIVOS ESPECIFICOS**

1. Manejar las operaciones básicas de entradas y salidas para realización de programas en C.
2. Utilizar herramientas de programación estructurada para el desarrollo de programas más complejos.
3. Reconocer la potencia del lenguaje C en la aplicación práctica en el área, para el control de dispositivos eléctricos.

### **V. - PRE - REQUISITO**

Algoritmo

### **VI. - CONTENIDO**

#### **6.1. Unidades programáticas**

1. Elementos de un programa en lenguaje C.
2. Operaciones básicas de entrada por teclado y salida por pantalla.
3. Sentencias de control.
4. Punteros.
5. Funciones.
6. Arrays y cadenas de caracteres.
7. Archivos.
8. Diseño del control de un dispositivo a través del puerto paralelo de la PC

#### **6.2. Desarrollo de las unidades programáticas**

1. Elementos de un programa en lenguaje C.
  - 1.1. identificadores y palabras claves.
  - 1.2. Tipos de datos simples.
  - 1.3. Constantes.
  - 1.4. Variables.
  - 1.5. Declaraciones.
  - 1.6. Sentencias.
  - 1.7. Expresiones y operadores.
  - 1.8. Funciones.
  - 1.9. Comentarios.
  - 1.10. Directivas del procesador de C.
2. Operaciones básicas de entrada por teclado y salida por pantalla.
  - 2.1. La función printf
  - 2.2. La función scanf
3. Sentencias de control.

- 3.1. La sentencia if.
- 3.2. La sentencia switch.
- 3.3. La sentencia while.
- 3.4. La sentencia do-while.
- 3.5. La sentencia for.
- 3.6. Bucles anidados.
  - 3.6.1. La sentencia break
  - 3.6.2. La sentencia continue.
  - 3.6.3. La sentencia goto.
4. Punteros.
  - 4.1. Declaración de punteros.
  - 4.2. Operaciones con punteros.
5. Funciones.
  - 5.1. Definición de una función
  - 5.2. Llamada a una función
  - 5.3. Declaración de una función
  - 5.4. Paso de parámetros a una función
  - 5.5. Reglas de ámbito de variables.
6. Arrays y cadenas de caracteres.
  - 6.1. Arrays unidimensionales.
  - 6.2. Arrays bidimensionales.
  - 6.3. Arrays multidimensionales.
  - 6.4. Cadenas de caracteres.
  - 6.5. Paso de arrays a funciones.
7. Archivos.
  - 7.1. Introducción
  - 7.2. Apertura de un archivo
  - 7.3. Cierre de un archivo
  - 7.4. Control de final de archivo.
  - 7.5. Escritura y lectura de un archivo.
  - 7.6. Acceso secuencial
  - 7.7. Acceso directo.
8. Diseño del control de un dispositivo a través del puerto paralelo de la PC: utilizando lenguaje C.
  - 8.1. Presentación del puerto paralelo de la PC
  - 8.2. Modos de operación del puerto paralelo.
  - 8.3. Direcciones de I/O del puerto paralelo
  - 8.4. Registros de datos, estados y de control.
  - 8.5. Instrucciones para la lectura y escritura del puerto paralelo.
  - 8.6. Presentación del proyecto a realizar: HARDWARE y SOFTWARE.

## VII. - ESTRATEGIAS METODOLÓGICAS

1. Presentación de la parte teórica con diferentes técnicas.
2. Resolución de ejercicios en lenguaje C y pruebas en la computadora.
3. Resolución de ejercicios aplicando las herramientas de programación y pruebas en la computadora.
4. Elaboración y presentación de trabajos prácticos de programación.
5. Realización y presentación de proyectos de programas de control (en lenguaje C), para captación de señales externas y activación de salidas a través del puerto paralelo de la PC.

## VIII. - MEDIOS AUXILIARES

1. Pizarrón
2. Equipo multimedia.
3. Resúmenes.
4. Interfase electrónica conectada al puerto paralelo de la PC para captación de señales externas y activación de salidas.
5. Material bibliográfico.

## IX. - EVALUACIÓN

- El estudiante deberá presentarse a dos Exámenes Parciales. Podrá presentarse al Tercer Examen Parcial el estudiante que haya obtenido un promedio inferior a 50% en los dos primeros exámenes parciales o que no se haya presentado en uno de ellos. Bajo esta situación, el promedio se realizará con las dos mejores puntuaciones.
- El promedio de los exámenes parciales será uno de los requisitos que habilite para el Examen Final, de acuerdo con la siguiente escala:
  1. Promedio igual o mayor a sesenta por ciento (60%), a partir del Primer Examen Final.
  2. Promedio igual o mayor a cincuenta por ciento (50%), a partir del Segundo Examen Final.
  3. Promedio inferior a 50%, el estudiante deberá volver a cursar la asignatura.

- Para tener derecho al Examen Final, el estudiante deberá cumplir con lo siguiente:
  1. Haber aprobado las asignaturas pre-requisitos.
  2. Tener el promedio habilitante.
  3. Cumplir con el porcentaje de asistencia mínimo, conforme a lo estipulado en la Planilla de Cátedra.
  4. Otros requisitos exigidos por la Cátedra, establecidos en la Planilla de Cátedra.

## X. - BIBLIOGRAFÍA

### MATERIALES BIBLIOGRÁFICOS DISPONIBLES EN LA BIBLIOTECA DE LA FACULTAD POLITÉCNICA

- Burns, A. & Andy W. (2003). *Sistemas de tiempo real y lenguajes de programación*. (3° Ed.). Madrid: Addison Wesley.
- Gallego León, J. (1998). *Técnicas de programación*. Madrid: McGraw-Hill.
- Hernández, R., Carlos Lázaro, J., Dormido, R. & Ros, S. (2001). *Estructuras de datos y algoritmos*. Madrid: Prentice Hall.
- Joyanes Aguilar, L. & Sánchez García, L. (2006). *Programación en C++: un enfoque práctico*. Madrid: McGraw-Hill.
- Joyanes Aguilar, L. (2003). *Fundamentos de programación: algoritmos, estructuras de datos y objetos*. (3° Ed.). Madrid: McGraw-Hill.
- Joyanes Aguilar, L., Castillo Sanz, A., Sánchez García, L. & Zahonero Martínez, I. (2005). *C, algoritmos, programación y estructura de datos*. Madrid: McGraw-Hill.

### RECURSOS DISPONIBLES A TRAVÉS DE CICCO

- Bellot, A. C., Pascual, F. L., Largo, F. L., Domenech, M. M., Lizán, F. M., Ortín, F. O., &... Cuerda, R. S. (2001). *Capítulo 6: Lenguajes de Programación*. (Spanish). *Fundamentos De Programación*. Vol. I. Metodología, 161. Recuperado de: <http://eds.b.ebscohost.com>
- Carlos, P., & Alonso, E. (2007). *Ruby: Lenguaje de Programación para Sistemas Distribuidos*. *Conciencia Tecnológica*, Iss 33, Pp 81-83 (2007), (33), 81. Recuperado de: <http://eds.b.ebscohost.com>
- Corbí Bellot, A. (2001). *Fundamentos de programación*. [Alicante]: Digitalia. Recuperado de: <http://eds.b.ebscohost.com>
- Tito Flórez, C. (2011). *Introducción a los microcontroladores RISC en Lenguaje C. PIC's de Microchips*. *Ingeniería E Investigación*, Vol 0, Iss 45, Pp 40-46 (2011), (45), 40. Recuperado de: <http://eds.b.ebscohost.com>