



<b>Nombre de la materia:</b>	Métodos Numéricos
<b>Clave:</b>	CB0300-T
<b>No. de horas/semana:</b>	4
<b>Total de horas:</b>	64
<b>No. de créditos:</b>	8
<b>Prerrequisitos:</b>	Programación de Computadoras (CI0000-T), Cálculo I ( CB0000-T), Álgebra Superior ( CB0200-T)

**Objetivo general:** El estudiante comprenderá las ideas en las que se basan los métodos numéricos de mayor uso en la solución de problemas de ingeniería, la estimación de los errores implicados en cada método y realizará la codificación de los algoritmos de cada método numérico visto en clase o alguna variante de éstos en el lenguaje seleccionado a partir del pseudocódigo explicado en clase por el profesor.

#### Programa sintético

1. Introducción a la programación en el lenguaje a utilizar. ....	4 hrs.
2. Modelos, computadoras y errores. ....	4 hrs.
3. Solución de ecuaciones no lineales .....	12 hrs.
4. 1er Examen parcial .....	2 hrs.
5. Solución de sistemas de ecuaciones lineales y no lineales .....	6 hrs.
6. Optimización .....	10 hrs.
7. Ajuste de curvas .....	8 hrs.
8. 2do Examen parcial .....	2 hrs.
9. Diferenciación e integración .....	6 hrs.
10. Ecuaciones diferenciales ordinarias .....	8 hrs.
11. 3er Examen parcial .....	2 hrs.
Total: 64 hrs.	

#### Programa desarrollado

1. Introducción a la programación en el lenguaje a utilizar. ....	4 hrs.
1.1 Introducción (Constantes, Variables, Operadores, funciones predefinidas)	
1.2 Instrucciones secuenciales	
1.3 Instrucciones condicionales	
1.4 Instrucciones de repetición	
1.5 Manejo de matrices y vectores	



- 1.6 Estructura de scripts y funciones
2. Modelos, computadoras y errores. .... 4 hrs.
- 2.1 Introducción
- 2.2 Modelos y computadoras. Ejemplo: Segunda Ley de Newton del Movimiento, modelo de la Caída de un paracaidista.
- 2.3 El concepto de error y su estimación. Errores de redondeo y errores de truncamiento. Error exacto y error estimado
- 2.4 Exactitud y precisión. Cifras significativas. El criterio de Scarborough.
- 2.5 Ejemplo: Serie de Taylor y errores de truncamiento. Comparativo del error estimado contra el error exacto.
3. Solución de ecuaciones no lineales ..... 12 hrs.
- 3.1 Introducción
- 3.2 Método gráfico. Solución e interpretación de la misma a través de las capacidades de graficación del lenguaje utilizado
- 3.3 Método de bisección. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 3.4 Método de la falsa posición (regula falsi). Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 3.5 Método de iteración de punto fijo. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/ código para implementar en el lenguaje utilizado.
- 3.6 Problemas de convergencia del método de punto fijo.
- 3.7 Método de Newton-Raphson . Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/ código para implementar en el lenguaje utilizado.
- 3.8 Problemas de convergencia del método de Newton-Raphson. Ventajas y desventajas con respecto a otros métodos.
- 3.9 Evaluación de polinomios Forma general del polinomio. Forma factorizada anidada del polinomio. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 3.10 Deflación de polinomios. Algoritmo. Seudocódigo/ código para implementar en el lenguaje utilizado.
- 3.11 División de polinomios. Algoritmo. Seudocódigo/ código para implementar en el lenguaje utilizado.
- 3.12 Solución de un polinomio de orden 2 en la forma general. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 3.13 Método de Bairstow para la solución de polinomios. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
4. 1er Examen parcial ..... 2 hrs.
5. Solución de sistemas de ecuaciones lineales y no lineales ..... 6 hrs.
- 5.1 Introducción.



- 5.2 Método de iteración de punto fijo para sistemas de ecuaciones. Análisis teórico de su funcionamiento. Contraste con el método de iteración de punto fijo para una sola ecuación. Algoritmo. Seudocódigo/ código para implementar en el lenguaje utilizado.
- 5.3 Método de Newton-Raphson para sistemas de ecuaciones. Análisis teórico de su funcionamiento. Contraste con el método de iteración de Newton-Raphson para una sola ecuación. La matriz Jacobiana y su evaluación. Algoritmo. Seudocódigo/ código para implementar en el lenguaje utilizado.
- 5.4 Método de Gauss-Jordan. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementación en el lenguaje utilizado.
6. Optimización ..... 10 hrs.
- 6.1 Introducción
- 6.2 Optimización no restringida. Método de Newton. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 6.3 Optimización no Restringida. Método de búsqueda de la sección dorada Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/ código para implementar en el lenguaje utilizado.
- 6.4 Optimización lineal restringida (programación lineal) Método Simplex. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
7. Ajuste de curvas ..... 8 hrs.
- 7.1 Introducción
- 7.2 Regresión lineal por el método de mínimos cuadrados. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 7.3 Regresión no Lineal mediante cambio de variable linealizante (modelos exponencial, potencia y modelo de la razón de crecimiento). Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 7.4 Regresión polinomial por mínimos cuadrados. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 7.5 Interpolación lineal. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 7.6 Interpolación cuadrática. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 7.7 Interpolación de Lagrange. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
8. 2do Examen parcial ..... 2 hrs.
9. Diferenciación e integración ..... 6 hrs.
- 9.1 Introducción
- 9.2 Diferenciación por diferencias divididas finitas atrás, adelante y centrales de exactitud simple. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.



- 9.3 Diferenciación por diferencias divididas finitas atrás, adelante y centrales de exactitud mejorada. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 9.4 Integración por el método de rectángulos. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 9.5 Integración por el método de trapezoides. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 9.6 Integración por el método de la regla de Simpson 1/3. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 9.7 Integración por el método de la regla de Simpson 3/8. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
10. Ecuaciones diferenciales ordinarias ..... 8 hrs.
- 10.1 Introducción
- 10.2 Integración por el método de Euler. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 10.3 Integración por el método de Heun con solo uno y con varios predictores (sin iteración e iterativo). Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado. Integración por el método del punto medio. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 10.4 Fórmulas de integración por algunos métodos de Runge-Kutta de segundo orden clásicos. Forma general para el cálculo de Kn. Algoritmos de Heun con un solo predictor, del punto medio y de Ralston. Seudocódigo/código para implementar en el lenguaje utilizado.
- 10.5 Fórmula de integración por el método de Runge-Kutta clásico de tercer orden. Confirmación de la forma general para el cálculo de Kn. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 10.6 Fórmula de integración por el método de Runge-Kutta clásico de cuarto orden. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
- 10.7 Solución de sistemas de ecuaciones diferenciales. Generalización de los métodos vistos para una sola ecuación diferencial al caso de sistemas de ecuaciones diferenciales.
- 10.8 Control del paso de integración en métodos de Runge-Kutta. Análisis teórico de su funcionamiento. Algoritmo. Seudocódigo/código para implementar en el lenguaje utilizado.
11. 3er Examen parcial ..... 2 hrs.

**Bibliografía básica:**

"Métodos Numéricos para Ingenieros". Chapra, Steven C.; Canale, Raymond P. McGrawHill.



### Bibliografía complementaria:

"Análisis Numérico".  
Burden, Richard L.; Faires, Douglas J.  
International Thomson Editores.  
"Análisis Numérico con Aplicaciones".  
Gerald, Curtis F.; Wheatley, Patrick O.  
Addison Wesley, 1999.  
"Métodos Numéricos con el lenguaje utilizado".  
Mathews, John H.; Fink, Kurtis D.  
Prentice Hall.  
"Métodos numéricos con Scilab"  
Héctor Manuel Mora Escobar.  
Editorial Universidad Central.

### Metodologías de enseñanza-aprendizaje:

- Revisión de conceptos, análisis y solución de problemas en clase ( X )
- Lectura de material fuera de clase ( X )
- Ejercicios fuera de clase (tareas) ( X )
- Investigación documental ( X )
- Elaboración de reportes técnicos o proyectos ( X )

### Metodologías de evaluación:

- Asistencia ( X )
- Tareas ( X )
- Elaboración de reportes técnicos o proyectos ( X )
- Exámenes de academia o departamentales ( X )

### Revisores:

Actualización propuesta por: Dr. José Juan Rincón Pasaye  
Fecha: 29 de junio de 2015

**Notas:** \*Nota: El lenguaje a utilizar a lo largo de este curso deberá podrá ser Matlab, o bien, un lenguaje de uso libre con características similares, como Scilab. Resumen de cambios de la propuesta de modificación del programa de Métodos Numéricos 1.- Porcentaje de cambios: 40% 2.- Se deja libre el uso de un lenguaje similar a Matlab, por ejemplo Scilab. 3.- Se hacen explícitos algunos temas que sólo



estaban implícitos, tales como: Errores estimados y errores exactos, errores de redondeo y de truncamiento, cifras significativas, precisión y exactitud, criterio de Scarborough. 4.- Se agrega el tema de regresión no lineal mediante cambio de variable linealizante y regresión polinomial por mínimos cuadrados, interpolación de Lagrange, solución de sistemas de ecuaciones diferenciales y control del paso de integración en métodos de Runge Kutta. Ya que se considera que son un pequeño paso adicional a los temas que ya se incluían. 5.- Se elimina el tema de interpolación de Fourier y todo el capítulo de solución de ecuaciones diferenciales parciales. Ya que se considera que son temas más especializados y que requieren haber cursado materias que no están en los prerrequisitos.