



DEPARTAMENTO DE ENSEÑANZA DE LA MATEMÁTICA  
MA-0560 COMPUTACIÓN Y MÉTODOS NUMÉRICOS  
CARTA AL ESTUDIANTE  
II CICLO 2018

**Horario:** Martes 10:00-11:50 Aula 212FM, Viernes 9:00-10:50 Aula 212FM

**Nivel:** IV año / II ciclo

**Requisitos:** MA-0550

**Tipo de curso:** Teórico

**Créditos:** 4

**Horas:** 4

## 1. Información general y descripción

Reciba la más cordial bienvenida al curso MA-0560: Computación y Métodos Numéricos. En este documento encontrará información sobre los aspectos del curso que usted debe conocer, tales como objetivos, contenidos, cronograma, metodología, evaluación y bibliografía.

Este curso está dirigido a estudiantes de octavo ciclo de la carrera Enseñanza de la Matemática y tiene como objetivo proporcionar las herramientas básicas del análisis numérico. En este curso se estudiará el concepto de aproximación por medio de los métodos numéricos y su programación. Además, se resolverán numéricamente problemas de diferente índole siguiendo las siguientes etapas: deducción matemática del algoritmo, implementación del algoritmo en un lenguaje de programación, aplicación y valoración del algoritmo programado en la resolución de problemas.

El curso Computación y Métodos Numéricos tiene una estrecha relación con varios de los cursos estudiados con anterioridad, pues integra diferentes conocimientos utilizados en la resolución de problemas que involucran, números reales, ecuaciones de una variable, estimación de fórmulas de funciones, derivación, integración, ecuaciones diferenciales. Además, fomenta el desarrollo del eje de tecnología y el eje investigativo. La resolución de problemas es actualmente un tópico sumamente importante en la enseñanza de la matemática.

Se requiere que el estudiante desarrolle su capacidad de pensamiento abstracto. Que obtenga conclusiones sobre cómo resolver un problema, reconociendo las hipótesis planteadas, y utilizar los conceptos teóricos en el planteamiento de la solución de dicho problema. Para este fin será necesario incluir algunas demostraciones simples y la generalización de algunos conceptos, sin llegar a un nivel de abstracción extremo. Este curso requiere que el estudiante dedique una buena cantidad de tiempo a comprender los diferentes conceptos y los resultados teóricos estudiados en la clase, teniendo al menos una dedicación de 7 horas de estudio independiente.

## 2. Objetivos generales

- Reconocer los principios básicos del análisis numérico.
- Comprender su importancia en la solución aproximada de problemas para los cuales no es posible o no resulta práctico encontrar soluciones exactas.
- Comparar la eficiencia y exactitud de los distintos métodos para un mismo problema.
- Incorporar las herramientas computacionales en la ejecución de algoritmos numéricos y conozcan las limitaciones de los mismos.

## 3. Objetivos específicos

- Dominar los fundamentos teóricos en que se basan los principales métodos numéricos del cálculo en una variable y el álgebra lineal.
- Demostrar el vínculo existente entre la teoría matemática estudiada y la solución de ejercicios prácticos que surgen en el ámbito de la ingeniería.
- Comprender los alcances y limitaciones teóricas asociadas a los métodos numéricos considerados en el curso.
- Adquirir los elementos básicos de uso y la programación en MATLAB y Mathematica.

## 4. Contenidos

La cantidad de semanas indicada para cada tema es una aproximación. Además, la semana previa a cada examen se dedicará un día a práctica por parte de los estudiantes en clase. El profesor puede invertir un tiempo distinto en cada tema según lo considere pertinente.

### 1. Elementos de programación en MATLAB (1 Semana):

#### 1.1 Ejecución condicional, ciclos.

Aplicar los conceptos básicos de la programación, uso de condicionales y ciclos en MATLAB. Estos incluyen: *if*, *for*, *while*, *do*.

#### 1.2 Manipulación de arreglos.

Manipular matrices y vectores usando MATLAB. Por ejemplo, estudiar como transponer, sumar (restar) y multiplicar matrices. Al igual que poder generar matrices usando ciclos y condicionales.

### 1.3 Matrices esparcidas en MATLAB\*

Manipular matrices con  $m \times n$  con la mayoría de sus entradas *ceros*.

## 2. Aritmética de precisión finita (1 Semana):

### 2.1 Punto flotante.

Representar un número en la computadora *punto flotante*.

### 2.2 Error absoluto, error relativo, dígitos significativos.

Medir el error de dicha representación en punto flotante (“error” producido por la representación). Calcular errores de las diferentes representaciones utilizadas.

### 2.3 Operaciones aritmética en precisión finita.

Estudiar las operaciones: suma, resta, multiplicación y división de las representaciones utilizadas. Poder aproximar la representación numérica luego de hacer las operaciones básicas.

### 2.4 Propagación del error.

Utilizar el truncamiento y redondeo para obtener valores representables al conjunto de números reales de precisión finita.

## 3. Solución numérica de ecuaciones no lineales (2 Semanas).

### 3.1 Método de Bisección

Aplicar el método de *Bisección* para aproximar la solución de una ecuación.

### 3.2 Método de Punto Fijo.

Aplicar el método del *Punto Fijo* para aproximar la solución de una ecuación.

### 3.3 Método de Newton-Raphson y Secante.

Aplicar los métodos de *Newton-Raphson* y *Secante* para aproximar la solución de una ecuación.

### 3.4 Aceleración de convergencia: método de Steffensen\*.

## 4. Métodos directos para la solución numérica de sistemas de ecuaciones lineales (2.5 Semanas).

### 4.1 Eliminación Gaussiana y Factorización LU.

Usar MATLAB para resolver sistemas de ecuaciones lineales utilizando los *métodos directos*: Eliminación Gaussiana y Factorización LU. Esto implica la programación e implementación de estos métodos en MATLAB.

#### 4.2 Factorización de Cholesky\*.

Definir y estudiar la factorización de Cholesky. Programar e implementar este método en MATLAB

#### 4.3 Matrices tridiagonales: Algoritmo de Thomas\*.

#### 4.4 Transformaciones de Householder y factorización QR.

Aplicar las transformaciones de Householder para encontrar la factorización QR. Programar e implementar en MATLAB.

### Parcial I

#### 5. Métodos iterativos para la solución numérica de sistemas de ecuaciones lineales (1.5 Semanas).

##### 5.1 Normas de vectores y matrices.

Definiciones y cálculo de Norma de vectores y Norma matricial. Al igual que la programación de estas utilizando MATLAB.

##### 5.2 Jacobi, Gauss-Seidel y métodos de relajación.

Aplicar los métodos iterativos: (*Jacobi*, *Gauss-Seidel* y *Relajación*) para la solución numérica de sistemas de ecuaciones lineales. Estudiar los criterios de parada usando el *error relativo*, *error absoluto*, y su programación e implementación en MATLAB.

##### 5.3 Métodos Iterativos tipo Richardson: Gradiente y Gradiente Precondicionado\*.

Aplicar los conceptos del método iterativo de (*Richardson*) y su programación e implementación en MATLAB.

#### 6. Aproximación de curvas mediante polinomios y segmentos de polinomios (2.5 Semanas).

##### 6.1 Interpolación de Lagrange, método de Newton.

Aplicar los métodos iterativos de (*Lagrange* y *Newton*), cota de error para el polinomio, y su programación e implementación en MATLAB.

##### 6.2 Método de Neville\*.

Aplicar los conceptos del método iterativo de (*Neville*) y su programación e implementación en MATLAB.

##### 6.3 Interpolación de Hermite\*.

Aplicar los conceptos del método iterativo de (*Hermite*), cota de error para el polinomio, y su programación e implementación en MATLAB.

#### 6.4 Interpolación cúbica segmentaria.

Aplicar los conceptos del método iterativo de (*Interpolación Cúbica*), estimado de error para el polinomio, y su programación e implementación en MATLAB.

#### 6.5 Interpolación paramétrica\*.

Aplicar los conceptos del método iterativo de (*Interpolación Paramétrica*), y su programación e implementación en MATLAB.

### Parcial II

#### 7. Diferenciación e Integración numérica (2.5 Semanas).

##### 7.1 Diferenciación numérica, fórmulas de 3 y 5 puntos de Newton.

Aplicar los conceptos de diferencias finitas de (*Newton*) para la aproximación de soluciones de ecuaciones diferenciales ordinarias y derivadas parciales. Al igual que su programación e implementación en MATLAB.

##### 7.2 Extrapolación de Richardson.

Aplicar los conceptos de la extrapolación de (*Richardson*), y su programación e implementación en MATLAB.

##### 7.3 Métodos básicos de integración numérica: punto medio, regla del rectángulo, trapecio, Simpson.

Aplicar los conceptos de integración numérica (*Regla del Trapecio, Regla de Simpson*) y su programación e implementación en MATLAB.

##### 7.4 Cuadraturas Gaussianas\*.

##### 7.5 Integración numérica compuesta.

Aplicar los conceptos de integración numérica (*Regla Compuesta del Trapecio, Regla Compuesta de Simpson*) y su programación e implementación en MATLAB.

##### 7.6 Integración de Romberg.

Aplicar los conceptos del método iterativo de (*Romberg*), estimado de error para el polinomio, y su programación e implementación en MATLAB.

#### 8. Métodos numéricos para problemas de valor inicial (3 Semanas).

##### 8.1 Métodos de Taylor de orden superior.

##### 8.2 Métodos Euler, Euler Modificado y Runge-Kutta.

Aplicar los conceptos de *Euler*, *Euler Modificado*, y *Runge-Kutta (de orden 2, 3, 4)* y su programación e implementación en MATLAB.

### 8.3 Interpolación de las soluciones\*.

### 8.4 Fórmulas multi-paso. Métodos de Adams-Moulton y Adams-Bashforth.

Aplicar *Fórmulas multi-paso* y métodos iterativos *Adams-Moulton*, y *Adams-Bashforth (de m-pasos)*. Programar en MATLAB.

### 8.5 Predictor-Corrector.

Aplicar los conceptos de los métodos iterativos (*Predictor-Corrector*) y su programación e implementación en MATLAB.

## Parcial III

**NOTA:** Los temas marcados con \* no serán vistos en clase y corresponderán a tareas a realizar por los estudiantes.

## 5. Cronograma

Semana	Fecha	Contenido y actividades
1	13-17 Agosto	1.1, 1.2 y 1.3
2	20-24 Agosto	2.1, 2.2, 2.3 y 2.4
3	27-31 Agosto	3.1 y 3.2
4	3-7 Setiembre	3.2 y 3.3
5	10-14 Setiembre	4.1
6	17-21 Setiembre	4.1 y 4.4
7	24-28 Setiembre	4.4 y 5.1
8	1 - 5 Octubre	5.1 y 5.2 - I Parcial
9	8-12 Octubre	6.1
10	15-19 Octubre	6.4
11	22-26 Octubre	6.4 y 7.1
12	29 Octubre - 2 Noviembre	7.2 y 7.3 - II Parcial
13	5-9 Noviembre	7.5 y 7.6
14	12-16 Noviembre	8.1 y 8.2
15	19-23 Noviembre	8.2 y 8.4
16	26-30 Noviembre	8.5 - Examen de Laboratorio
17	3-7 Diciembre	III Parcial
18	10-14 Diciembre	Ampliación y Suficiencia

## 6. Metodología

Entre las estrategias principales para desarrollar el curso están la clase magistral, trabajo individual y discusiones de temas. Para esta última es fundamental la participación del estudiante. En cada unidad didáctica se dedicarán lecciones al desarrollo teórico y práctico. En las lecciones prácticas es sumamente importante la participación del estudiante en la resolución de problemas, para lo cual es vital la dedicación que el estudiante le asigne a la práctica en sus horas de estudio en casa.

El estudiante requiere de muchas horas de estudio fuera de clase para trabajar los ejercicios propuestos, dado que en cada capítulo encontrará suficientes de ellos que ilustran los contenidos de las unidades didácticas. Por eso es necesario tener disponibilidad para dedicar horas de estudio fuera de clase.

Se dedicarán las clases de los viernes para trabajar en el laboratorio usando paquetes de software, principalmente: MATLAB y Mathematica.

## 7. Evaluación

La evaluación sumativa incluirá los siguientes rubros:

- Tres exámenes parciales de 20 % cada uno.
- Tareas y quices: 15 %. Las tareas se realizan en grupos de máximo 3 personas y se deben presentar por escrito una por grupo (se sugiere que usen latex para escribir la solución).
- Proyecto: 15 %. Este proyecto se trabajará en grupos y los temas serán repartidos al inicio del curso. Además, **se debe entregar un reporte escrito** del proyecto desarrollado. **Este rubro incluirá además la exposición en clases de los proyectos.**
- Examen de laboratorio: 10 %. Este se realizará el último día de clases y buscará poner a prueba lo aprendido por el estudiante para resolver problemas ayudado con la computadora.

Los exámenes se realizarán, tentativamente, las siguientes fechas:

Examen	Día	Hora	Contenido
Examen 1	Viernes 5 de Octubre	9am	Temas: 1, 2, 3 y 4
Examen 2	Viernes 29 de Octubre	9am	Temas: 5 y 6
Examen Laboratorio	Viernes 30 de Noviembre	9am	Toda la materia
Examen 3	Miércoles 5 Diciembre	8am	Temas: 7 y 8
Ampliación	Jueves 13 Diciembre	8am	Toda la materia

Para realizar examen de reposición, se debe entregar al profesor la solicitud por escrito acompañada con el documento oficial que justifique debidamente la razón de su ausencia al examen

respectivo, según las causas y periodos que el Reglamento de Régimen Académico Estudiantil considera como válidas. Una vez aprobada la reposición, el profesor le indicará al estudiante la fecha de reposición.

**Reporte de la nota final:** Para efectos de promoción rigen los siguientes criterios, los cuales se refieren a la nota de aprovechamiento NA del curso, expresada en una escala de 0 a 10, redondeada, en enteros y fracciones de media unidad, según el reglamento vigente:

- Si  $NA > 6.75$  el estudiante gana el curso con calificación NA redondeada a la media más próxima, los casos intermedios como 7.25 se redondean hacia arriba, es decir, 7.5.
- Si  $5.75 < NA < 6.75$ , el estudiante tiene derecho a realizar el examen de ampliación, en el cual se debe obtener una nota superior o igual a 7 para aprobar el curso con nota 7, en caso contrario su nota será 6.0 o 6.5, la más cercana a NA.
- Si  $NA < 5.75$  pierde el curso.

## 8. Referencias bibliográficas

El curso se basará principalmente en las notas compiladas por el profesor. Otro material destacado es el siguiente:

1. Blum E. K., *Numerical analysis and computation theory and practice*. Addison-Wesley, 1972.
2. Burden, R. & Faires, D., *Numerical Analysis*, Thomson Learning, 2002.
3. Catmull, E & Rom, R., *A class of local interpolating Splines*. En *Computer Aided Geometric Design*, Academic Press, 1974.
4. W. Ford. *Numerical linear algebra with applications: Using MATLAB*, Elsevier/Academic Press, Amsterdam, 2015.
5. Ralston A., *Introducción al análisis numérico*. Limusa, 1978.
6. Sheid, F., *Análisis Numérico*. Serie Shaum, McGraw Hill, 1968.
7. N. Trefethen y D. Bau. *Numerical Linear Algebra*. SIAM, 1997.

## 9. Atención a estudiantes

- Profesor: Esteban Segura Ugalde.
- Correo electrónico: ma0560ucr@gmail.com / esteban.seguraugalde@ucr.ac.cr
- Teléfono: 2511 3418.
- Oficina: 206 CIMPA (Nuevo edificio de Matemática, Ciudad de la Investigación).
- Horario de consulta en oficina: Martes 12-1, Viernes 11-1, o con cita previa.
- Horario de clase: Martes 10:00-11:50 Aula 212FM, Viernes 9:00-10:50 Aula 212FM