

# Información del curso

Pedro O. Pérez M., PhD.

Implementación de métodos computacionales  
Tecnológico de Monterrey

*pperezm@tec.mx*

02-2023

## 1 Información del profesor

Información del profesor

## 2 Información del curso

Competencias

Metodología

Evaluación

Normas de clase

Bibliografía

- Pedro Oscar Pérez Murueta
  - ISC Mayo 1994
  - MTI Mayo 2002
  - DCC Diciembre 2019
- Correo: [pperezm@tec.mx](mailto:pperezm@tec.mx)
- Zoom: <https://itesm.zoom.us/my/pperezm>
- Horario de asesoría: <https://shorturl.at/hptI9>



# STC0100 Desarrollo de algoritmos computacionales

Soluciona problemas generando algoritmos computacionales eficientes bajo modelos y herramientas de las ciencias computacionales.

Subcompetencia	Nivel de dominio
STC0101 Implementación de algoritmos computacionales - Implementa algoritmos computacionales confiables y correctos que solucionan problemas.	A
STC0102 Optimización de algoritmos computacionales - Optimiza algoritmos computacionales robustos y eficientes que se aplican en el desarrollo de soluciones.	A
STC0103 Generación de modelos computacionales - Genera modelos computacionales verificables y correctos para la solución de problemas.	A
STC0104 Implementación de modelos computacionales - Implementa modelos computacionales en la solución de un problema que permiten su interoperatividad con otros modelos.	A

Optimiza soluciones a las problemáticas de su ámbito profesional, con la incorporación inteligente y oportuna de tecnologías digitales de vanguardia.

Subcompetencia	Nivel de dominio
SEG0702 Tecnologías de vanguardia - Evalúa diversas tecnologías de la información, con apertura en la búsqueda e implementación de alternativas relevantes en la transformación de la práctica profesional.	B

- Esta materia consta de 5 partes (distribuidas en 5 módulos) que te permitirán conocer algoritmos avanzados, siguiendo en la línea de las estructuras de datos. Los contenidos detallados los puedes encontrar en la sección de contenidos.

- A lo largo de la materia deberás repetir una y otra vez esta secuencia:
  - ① Aprender conceptos teóricos y practicarlos haciendo ejercicios, de la misma forma que se hace en matemáticas.
  - ② Explorar su relación con la creación de los lenguajes de programación.
  - ③ Diseñar, basado en los conceptos, el modelo utilizado para resolver el problema.
  - ④ Implementar el diseño por medio de la creación de un programa computacional en un lenguaje de programación específico, cumpliendo con las indicaciones dadas por el instructor en cuanto a la entrada que recibe el programa y la salida que se espera que tenga.
  - ⑤ Hacer pruebas para verificar que es correcto y que cumple con lo solicitado.
  - ⑥ Preparar la documentación adecuada para el manejo del programa y ponerlo listo para su publicación.
- Una vez que el programa o la tarea estén listos se deberá entregar por el medio indicado, antes de la fecha límite establecida por el instructor.

- El instructor revisará el programa o la tarea entregada y te dará la retroalimentación adecuada para que sepas qué se hiciste bien y qué faltó, para incluirlo en la siguiente entrega.
- Algunas de estas actividades serán hechas en forma individual, pero otras tendrás que hacerlas en equipos de trabajo.
- Cada uno de estos desarrollos recibirán una serie de puntos los cuales conformarán tu calificación final. Además, cada una de estas actividades te permitirá desarrollar una serie de competencias necesarias para cualquier científico de la computación.



<b>Actividades</b>	<b>Evidencia de dominio</b>
Ejercicios de conceptos formales (2)	12 %
Actividades de programación (4)	35 %
E1. Resaltador de sintaxis (Integradora)	15 %
E2. Resaltador de sintaxis paralelo (Integradora)	15 %
E3. Análisis de herramientas usadas para la solución de las situaciones problema y su aportación al proyecto (Reflexión)	8 %
E4. Póster y presentación de las herramientas usadas para la solución de las situaciones problema (Reflexión)	15 %

## Calificaciones

- Las calificaciones se expresan en escala de uno a cien.
- La calificación mínima aprobatoria es 70 (SETENTA).

## Asistencia a clases

En lo que respecta a esta clase:

- La sesión de clase inicia 5 minutos después del horario establecido (9:05).

## Tareas

- Toda tarea tendrá su fecha y horario de entrega que es inamovible. Vencido el término de entrega no se recibirán tareas y/o proyectos.
- Todas las tareas son individuales a menos que explícitamente se pida trabajar en grupo.

## Redacción y Organización

- La mala redacción, organización y ortografía en la elaboración de tareas, proyectos, presentaciones y exámenes, será causa de penalización en la calificación correspondiente.

## Faltas a la Integridad Académica en Tareas, Proyectos o Exámenes

- Las faltas a la integridad académica, como la copia o tentativa de copia en cualquier tipo de examen o actividad de aprendizaje; el plagio parcial o total; facilitar alguna actividad o material para que sea copiada y/o presentada como propia; la suplantación de identidad; falsear información; alterar documentos académicos; vender o comprar exámenes o distribuirlos mediante cualquier modalidad; hurtar información o intentar sobornar a un profesor o cualquier colaborador de la institución; entre otras acciones más son consideradas faltas grave. Cuando un alumno cometa un acto contra la integridad académica, se le asignará una calificación reprobatoria a la actividad, examen, período parcial o final. La calificación reprobatoria asignada por el profesor será inapelable, y a esta sanción se sumarán las otras posibles que determine el Comité de Integridad Académica de Campus. Esto tal como lo indica el Reglamento Académico en su CAPÍTULO IX: Faltas a la integridad académica.

## Baja de materias

- Artículo 4.8 Los alumnos podrán darse de baja en una o varias unidades de formación inscritas durante el periodo académico semestral o periodos intensivos, **antes del último día de clases de la unidad de formación a dar de baja, conforme lo establezca el Calendario Escolar correspondiente.** Las unidades de formación dadas de baja no se registrarán como reprobadas. En el caso de que el alumno solicite la baja de todas sus unidades de formación, deberá pasar nuevamente por un proceso de admisión, en el que se tomará en cuenta su historial académico para la decisión. El proceso relacionado con la baja de las unidades de formación se llevará a cabo de acuerdo con las políticas y procedimientos administrativos definidas por la Dirección de Servicios Escolares para ese fin.

[Recursos base](#)

[Recursos de consulta](#)

## Recursos base



**Nombre:** Languages and machines: an introduction to the theory of computer science

**Autor:** Thomas Sudkamp

**Edición:** 3rd

**Año:** 2016

**Editorial:** Addison-Wesley

**ISBN:** 9780321322210



**Nombre:** Programming languages: principles and practice

**Autor:** Kenneth Louden

**Edición:** 3rd

**Año:** 2011

**Editorial:** Cengage Learning

**ISBN:** 9780321322210



[Racket Programming Language](#) ➞



[Clojure Programming Language](#) ➞

- 1 Horario de asesorías: <https://shorturl.at/hptI9>
- 2 Repositorio en Github: <https://github.com/Manchas2k4/tc2037.git>
- 3 Zoom: <https://itesm.zoom.us/my/pperezm>
- 4 Juez en línea: <https://vjudge.net/>
- 5 IDE: Atom, Sublime, Visual Studio Code, cualquier IDE que les guste.
- 6 Racket: <https://racket-lang.org>
- 7 SWI-Prolog: <https://www.swi-prolog.org/>