



Universidad de Costa Rica
Facultad de Ingeniería
Escuela de Ingeniería
Eléctrica

**IE-0117 Programación Bajo
Plataformas Abiertas**

EIE

Escuela de
Ingeniería Eléctrica

Profesor: Luis Daniel Ferreto Chavarría

ldanielfch@gmail.com

Horario: K, V: 7:00am - 10:00am.

Consulta: M: 5:30pm - 8:30pm (con previo aviso, se puede convenir hora fuera de este horario).

Modalidad Virtual.

Sigla: IE-0117

Ciclo: I

Horas lectivas: 3 horas de teoría y 3 horas de laboratorio.

Clasificación: Propio

Descripción

El curso de programación bajo plataformas abiertas está diseñado para enseñar a las personas estudiantes de Ingeniería Eléctrica, un conjunto de conocimientos de tipo teórico-prácticos, con el fin de brindar la preparación para abordar y resolver problemas ingenieriles, mediante el uso de herramientas y el desarrollo de programas de aplicación utilizando plataformas abiertas. El curso se enfoca en dos objetivos. Por un lado, la instalación, utilización y administración de plataformas abiertas, en particular, sistemas operativos GNU/Linux (Ubuntu). En segunda instancia, el aprendizaje de lenguajes de programación multipropósito, en particular C. Al finalizar el curso, las personas estudiantes estarán preparadas para incursionar en temas más avanzados de programación que les permitirá desarrollar soluciones a problemas afines a la carrera.

Objetivo General

Brindar a la persona estudiante los conocimientos y herramientas que le permitan resolver problemas ingenieriles mediante la instalación, utilización, administración y desarrollo de aplicaciones de software bajo plataformas abiertas.

Objetivos Específicos

Al finalizar el curso, el estudiante estará en capacidad de:

1. Conocer las características principales de las plataformas abiertas.
2. Instalar y configurar sistemas operativos GNU/Linux.
3. Desarrollar destrezas básicas del sistema operativo GNU/Linux.

4. Administrar y utilizar sistemas operativos GNU/Linux.
5. Desarrollar habilidades estructuradas de programación
6. Conocer diferentes paradigmas de programación y sus respectivos nichos de aplicación.
7. Desarrollar la habilidad de programación en el lenguaje C.
8. Implementar aplicaciones de software mediante el uso de diversas bibliotecas.
9. Aprender a gestionar la memoria dinámica en un programa
10. Desarrollar la capacidad para trabajar en equipo.
11. Desarrollar la capacidad para presentar oralmente.

Requisitos

- CI-0202 Principios de Informática

Recursos necesarios para el curso (debido a la modalidad Virtual):

- Computadora personal con las siguientes características mínimas:
 - Procesador de 2GHz, dual-core.
 - 4 gigabytes de memoria RAM.
 - 50 gigabytes de disco duro.
 - Sistemas operativos recomendados: Ubuntu 18.04, Ubuntu 20.04, Ubuntu 22.04 o Windows 10/11 (con máquina virtual para usar Linux). Se impartirán tutoriales para la instalación del sistema operativo utilizado en el curso.
- Conexión a internet.

Si el estudiante no cumple con alguno de estos requisitos debe coordinar una solución con el profesor.

Número de créditos

- 3 créditos

Metodología

El curso será impartido en dos módulos, donde cada uno de ellos contará con una serie de exposiciones teóricas y sus prácticas correspondientes. Se llevarán a cabo proyectos de programación, en los cuales el estudiante aprenderá a desarrollar las aplicaciones y a utilizar bibliotecas de soporte. Al estudiante se le solicitará proyectos de investigación bibliográfica, diseño e implementación, con la respectiva presentación oral y escrita (informe técnico) de sus resultados.

Se utilizará la plataforma de Mediación Virtual para poner a disposición los videos, clases en línea, documentos, y evaluaciones necesarias para el curso.

Todos los proyectos de programación deberán tener un repositorio Git, y las presentaciones se harán mediante presentación oral (en clases sincrónicas) o con un video corto del funcionamiento de este. Los enlaces a los repositorios y videos serán entregados mediante Mediación Virtual.

Durante los exámenes se les solicitará el uso de cámara y audio.

Sobre el hostigamiento y la discriminación

La Universidad de Costa Rica es un espacio de cero tolerancia al hostigamiento y la discriminación. En la Escuela contamos con un grupo de apoyo al que cualquier persona puede acercarse de forma voluntaria si siente que ha sido víctima de hostigamiento o discriminación, para obtener una guía sobre los recursos institucionales disponibles a los que puede acudir:

Recursos sobre hostigamiento y discriminación

Sobre el plagio

El plagio está indicado como una falta grave en el Reglamento de Orden y Disciplina de los Estudiantes, e implica una sanción de suspensión entre seis meses y seis años para quien lo cometa. Si bien debería ser parte del marco ético de cada persona no cometerlo, en el curso contamos con distintas técnicas y herramientas para identificarlo.

Evaluación

Proyecto	30%
Exámenes Cortos	15%
Laboratorios	25%
Tareas	10%
Examen Final	20%

1. Cada evaluación tendrá lineamientos distintos, indicados por sus enunciados.
2. El Reglamento de Régimen Académico Estudiantil incluye los lineamientos para reposiciones y reclamos en las actividades que aportan a la calificación final, entre otros.
3. No se permiten entregas tardías.
4. Respecto a los laboratorios:
 - a. Los laboratorios serán asignados durante la clase, y se contará con una semana para realizarlos a partir de la entrega del enunciado. Si bien el estudiante contará con tiempo dentro de la clase para la realización del mismo, se espera que el estudiante lo complete fuera de clase.
 - b. Cada laboratorio deberá venir con su respectivo reporte, de lo contrario no será revisado.
 - c. Los reportes deberán ser escritos en tercera persona con forma impersonal. Todas las figuras, tablas, ejemplos de código, e ilustraciones deberán venir etiquetadas y citadas en el texto. De requerirse, también deberán ser incluidas las referencias bibliográficas utilizadas. Todos los reportes deben incluir portada. NO se permite utilizar fotografías de la pantalla, en su lugar se deben usar capturas de pantalla hechas dentro de la misma computadora (en inglés: "screenshots"). Cualquier captura de pantalla debe ser recortada para incluir solamente la información necesaria para el reporte o tarea (por ejemplo: si se toma la captura de una terminal, se debe incluir sólo la terminal, y no la pantalla completa). Se debe cuidar la presentación, ortografía, redacción, calidad y letra de los materiales que presente.

5. Respecto a los exámenes cortos:
 - a. Estos serán realizados en el transcurso de la clase, a través de mediación virtual.
6. Respecto a las tareas,
 - a. Con el fin de incentivar la investigación por parte de los estudiantes, se les asignará diferentes temas sobre los cuales podrán investigar y exponer sus resultados ante la clase. La fecha de entrega de la misma no se especifica en el cronograma puesto que cada estudiante tendrá la libertad de elegir la fecha en la que presente su investigación.
7. Respecto al proyecto final:
 - a. El proyecto final podrá ser realizado de manera grupal o individual, a elección del estudiante. El mismo deberá ser realizado en el transcurso de varias semanas, y contará con un entregable a través de mediación virtual, así como una presentación de resultados que deberá ser realizada durante el horario de la clase.
8. Respecto al examen final:
 - a. El enunciado para el examen final será entregado a los estudiantes, y estos contarán con una ventana de tiempo para realizarlo.

Temas

Módulo 1: Plataformas abiertas: GNU/Linux

- A. Introducción: Historia, comunidad, filosofía y distribuciones
- B. Instalación y configuración básica de GNU/Linux como una máquina virtual y/o real usando distintas distribuciones (Debian, Ubuntu, entre otras).
- C. Herramientas: firefox, gimp, gedit, libreoffice, etc.
- D. Sistema de archivos, manejo de medios de almacenamiento y particiones.
- E. Uso del shell y otras herramientas de línea de comandos.
- F. Administración de usuarios, grupos y permisos.
- G. Manejo de procesos y servicios.
- H. Paquetería.

H. Control de versiones (git).

Módulo 2: Programación en un lenguaje imperativo (C)

1. Introducción al lenguaje de programación: historia, filosofía, compilador, entre otros.
2. Paradigmas de lenguajes de programación: imperativo (C) , orientado a objetos (Python/C++), funcional (Lisp), lógico (Prolog), etc
3. Características del lenguaje: tipos de datos, estructuras de control, etc.
4. Entorno de programación: Compiladores y herramientas para el desarrollo de aplicaciones (por ejemplo gcc, IDEs, etc.).

5. Automatización de proyectos: make, cmake, autotool.
6. Funciones y recursión.
7. Entrada y salida de archivos, manejo de archivos, funciones de sistema.
8. Punteros, punteros dobles, y manejo de memoria dinámica (Stack, Heap).
9. Structs.
10. Enums.
11. Introducción a las estructuras de datos
12. Lista enlazada simple y doblemente enlazada

Mediación Virtual

Este curso es virtual. Se utilizará la plataforma institucional Mediación Virtual para colocar los documentos y vídeos del curso. Además, se usará para realizar tareas, quices, laboratorios, y proyectos.

Cronograma

Fecha (semana)	Descripción de la clase	Fechas tentativas de evaluación
10 de marzo	Módulo 1: GNU/Linux	
17 de marzo		Quiz #1: 20 de marzo
24 de marzo		Entrega Lab #1: 24 de marzo
31 de marzo		Quiz #2: 3 de abril
7 de abril		Entrega Lab #2: 7 de abril
14 de abril		
21 de abril	Módulo 2: C	Entrega Lab #3: 21 de abril
28 de abril		Quiz #3: 1 de mayo
5 de mayo		Entrega Lab #4: 5 de mayo
12 de mayo		Quiz #4: 15 de mayo
19 de mayo		

26 de mayo		Propuesta proyecto final
2 de junio		Quiz #5: 5 de junio
9 de junio		Entrega Lab #5: 9 de junio
16 de junio		
23 de junio		
30 de junio		Entrega y presentacion Proyecto Final: 30 de junio
7 de julio	Semana de evaluaciones finales	Examen Final

Bibliografía

1. The Linux Documentation Project. <http://tldp.org/>.
2. IBM developerWorks. <http://www.ibm.com/developerworks/>.
3. McCarty, Bill. Learning Debian Gnu/Linux. Sebastopol, California: O'Reilly & Associates, 1999.
4. Wall, K. y Watson, M. Linux Programming Unleashed. Segunda Edición. Indianápolis: SAMS, 2000.
5. Hunger, S. Debian GNU/Linux Bible. New York: John Wiley and Sons, Inc., 2001.
6. Tackett Jr., J.; Gunter, D. Utilizando Linux. Segunda edición. Prentice Hall. 1996.
7. Kernighan, Brian y Pike, Rob. El entorno de programación de Unix. Prentice Hall, 1984.
8. Kernighan, Brian y Ritchie, Dennis. The C programming language. Segunda Edición. Prentice Hall, 1988.
9. Brian Kernighan, Dennis Ritchie El Lenguaje de Programación C. Prentice Hall.
10. Schildt, H. C/C++ Programmer's Reference. Segunda edición. New York: McGraw-Hill Osborne Media, 2000.
11. Yedidyah Langsam, Moshe J. Augenstein Estructuras de datos con C y C++ Prentice Hall,segunda edición
12. Schildt, H. C/C++ Programmer's Reference. Segunda edición. New York: McGraw-Hill Osborne Media, 2000.