



UNIVERSIDAD DE SONORA
Unidad Regional Centro
División de Ingeniería
Departamento de Ingeniería Industrial
LICENCIATURA INGENIERÍA EN MECATRÓNICA

Nombre de la Asignatura: ROBÓTICA OS

Clave:	Créditos: 8	Horas totales: 80	Horas Teoría: 2	Horas Práctica: 4	Horas Semana: 5
---------------	-----------------------	-----------------------------	---------------------------	-----------------------------	---------------------------

Modalidad: Presencial **Eje de formación:** Especializante

Elaborado por: DR. CARLOS FIGUEROA NAVARRO

Antecedente: **Consecuente:**

Carácter: Optativa **Departamento de Servicio:** Ingeniería industrial

Propósito:

La asignatura pertenece al eje especializante, y es de carácter optativa. El principal propósito es proporcionar a los estudiantes los aspectos fundamentales y básicos del desarrollo de software en forma de sistemas operativos para operar y programar robots y manipuladores. Aprender las diferentes metodologías de programación tanto por guiado como la programación textual.

I. Contextualización

Introducción:

El proceso de programación de un robot consiste en introducir en su sistema de control las instrucciones necesarias para que desempeñe las tareas para las que ha sido diseñado. Existen varios procedimientos de programación de robots.

1. Programación guiada o directa

El operario interviene guiando manualmente el brazo del robot, y hace que este vaya describiendo los movimientos y trace las trayectorias necesarias para cumplir su función. Cada uno de los movimientos realizados se va almacenando en la memoria del robot, de forma que podrán ser repetidos posteriormente, ya sin intervención humana. En este tipo de programación es necesario disponer del propio robot para la elaboración del programa.

2. Programación textual o indirecta

En este caso no es necesaria la presencia del robot para realizar el programa, puesto que este se lleva a cabo en un lenguaje informático. El programa consiste en un texto formado por un conjunto de instrucciones; cuando el programa sea grabado en la memoria del robot, este realizará las acciones indicadas en el mismo.

Este tipo de programación permite realizar operaciones más complejas y con mayor grado de precisión. Además, presenta la ventaja de que es posible establecer relaciones entre el robot y su entorno. Para ello basta con introducir en el programa los datos procedentes de los sensores de forma que el robot actúe en consonancia con los mismos, tal y como ocurre en los denominados «robots inteligentes».

A su vez, la programación textual puede ser de dos tipos: explícita y especificativa

La programación textual explícita se corresponde con los llamados lenguajes estructurados. Consiste en programar de forma secuenciada y estructurada el conjunto de acciones que debe realizar el robot para llevar a cabo la tarea encomendada. En dichas instrucciones pueden introducirse también las características del medio.

La programación textual especificativa está más en consonancia con los lenguajes de programación orientados a objetos. En este caso, el programa gira en torno a los elementos manipulados por el robot y las acciones que ha

de realizar con ellos, teniendo en cuenta el ámbito en el que se desarrollan dichas acciones.

Para programar un robot se sigue un proceso semejante al de la elaboración de un programa informático destinado a cualquier otra aplicación. Primero será necesario establecer el algoritmo idóneo que permita al robot llevar a cabo las tareas para las que ha sido diseñado, tras lo cual se traducirá dicho algoritmo en un lenguaje de programación inteligible por el sistema de control del robot. Dicho lenguaje debe permitir especificar de forma clara y sencilla las tareas que debe realizar el robot.

En esta materia también se estudia el sistema operativo robótico (ROS) que es una infraestructura digital (framework) para el desarrollo de software para robots que proporciona la funcionalidad de un sistema operativo. El desarrollo de este concepto es del año 2007. La idea surge en el Laboratorio de Inteligencia Artificial de Stanford para dar soporte al proyecto del robot con inteligencia artificial de Stanford. Actualmente las investigaciones y el desarrollo se continúan en EEUU en el instituto Willow Garage, que es una incubadora de empresas y un laboratorio de investigación robótica dedicada a la creación de software de código abierto para las aplicaciones en robots.

ROS tiene dos partes básicas: la parte del sistema operativo, ROS y ROS-PKG, que es un conjunto de paquetes aportados por la contribución de usuarios que implementen tareas complementarias o funcionalidades tales como localización y mapeo simultáneo, planificación, percepción, simulación.

ROS es software libre bajo términos de licencia BSD. Esta licencia permite libertad para uso comercial e investigador. Las contribuciones de los paquetes en ROS-PKG están bajo una gran variedad de licencias diferentes.

Las unidades didácticas se describen a continuación:

La Unidad didáctica I trata sobre el lenguaje Python que es un lenguaje interpretado, lo que facilita el proceso de programación. Además, Python provee una gran biblioteca de módulos para hacer toda clase de tareas. Dado que soporta tanto programación procedural como orientada a objetos, sirve como base para introducir conceptos importantes de informática como por ejemplo abstracción procedural, estructuras de datos, y programación orientada a objetos, que son aplicables a otros lenguajes como Java o C++.

En la unidad II describe el robot y los accesorios necesarios para trabajar con él y luego, se estudia la forma de interactuar con Python. La idea es usar los mecanismos más sencillos y luego incorporar otros.

Un tipo de robot a estudiar es el denominado robot Múltiple N6, que es un robot educativo extensible basado en la plataforma de prototipado Arduino. Tanto el hardware como el software utilizado son libres, es decir que tanto las especificaciones de la electrónica necesaria como la de los programas necesarios para su utilización están accesibles y pueden utilizarse libremente.

La Unidad didáctica III trata aspectos generales del ROS y algunas aplicaciones. Se estudia este sistema operativo que es una infraestructura digital (framework) para el desarrollo de software para robótica que proporciona la funcionalidad de un sistema operativo

En la Unidad didáctica IV se presenta un historial de versiones. El alumno conoce los modelos de éxito de robots con aplicación de un sistema operativo. Se presenta gran cantidad de modelos que tienen interés didáctico para que los alumnos estimulen su creatividad tecnológica.

En la Unidad didáctica V se estudian los robots y el hardware relacionado. Se presentan las principales características de la arquitectura hardware-software para el control de un robot. La arquitectura hardware tiene un sistema operativo para que todas las funciones del vehículo se controlan mediante sistemas de adquisición de datos modulares y controladores de hardware de interfaces de sensores y actuadores. La arquitectura debe ser probada para obtener buenos resultados, controlando el desempeño del robot y la gestión de procedimientos de captura, procesamiento y entrega de toda la información sensorial.

Perfil del(los) instructor(es):	Poseer Licenciatura en Ingeniería en Mecatrónica. Preferentemente con grado académico de maestría o especialidad. Con experiencia docente y desarrollo profesional comprobada cuando menos de dos años en el campo de la materia.
--	---

II. Competencias a lograr

Competencias genéricas a desarrollar:

- **Capacidad de aprender y actualizarse permanentemente.** Articula saberes de diversos campos y establece relaciones entre ellos y su vida cotidiana.
 - **Trabajo colaborativo.** Asume una actitud constructiva, congruente con los conocimientos y habilidades con los que cuenta dentro de distintos equipos de trabajo.
 - **Capacidad para la toma de decisiones.** Evalúa y sopesa información importante para identificar los aspectos relevantes. Define la prioridad para la solución del problema en términos de impacto y urgencia.
 - **Capacidad para realizar investigación básica y aplicada.** Identifica los sistemas y reglas o principios medulares que subyacen a una serie de fenómenos.
- Competencia Digital.** Aplica herramientas digitales para el pensamiento reflexivo, la creatividad y la innovación.

Competencias específicas:

- HABILIDAD PARA DESARROLLAR SISTEMAS DE CONTROL Y DE AUTOMATIZACIÓN
 - Definir conocimientos y obtener habilidades donde use los fundamentos del sistema operativo de robots. Aplicar los conocimientos aprendidos por medio de plataformas basadas en software comercial. Desarrollar aplicaciones.

Objetivo General:

Conocer las plataformas disponibles en el universo de la investigación robótica para la elaboración de sistemas operativos para robots y manipuladores, así como poder aplicar los desarrollos como software libre, conocimientos y experiencias a la programación de un prototipo primitivo. Aprender las diferentes metodologías de programación tanto por guiado como la programación textual.

Objetivos Específicos:

1. Conocer el lenguaje Python y su relación a programar robots
2. Aprender a interactuar el lenguaje Python con los robots en un ambiente Arduino.
3. Conocer los sistemas operativos enfocados a la robótica.
4. Establecer la clasificación del historial de versiones modelos de ROS.
5. Conocer el hardware de robots desarrollados. Elaborar un prototipo.

Unidades Didácticas:

Unidad Didáctica I – LENGUAJE PYTHON

Unidad Didáctica I- PROGRAMAR ROBOTS EN PLATAFORMA ARDUINO

Unidad Didáctica III- SISTEMA OPERATIVO PARA ROBOTS

Unidad Didáctica IV – HISTORIAL DE VERSIONES

Unidad Didáctica V – ROBOTS Y HARDWARE RELACIONADO

III. Didáctica del programa

Unidades Didácticas:

Unidad didáctica I. Lenguaje Python.

En la unidad I, el alumno adquiere conocimientos sobre Python que es un lenguaje de programación de alto nivel cuya filosofía hace hincapié en una sintaxis que se orienta a obtener un código legible. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional.

- Sintaxis
- Capacidad de introspección
- Orientación a objetos
- Expresión natural de procedimientos
- Modularización y soporte jerárquico de paquetes
- Manejo de errores basado en excepciones
- Tipos de datos dinámicos de muy alto nivel
- Extensas librerías estándar y módulos.

Unidad didáctica II. Programación de Robots

En la unidad I, el alumno adquiere conocimientos sobre los elementos para programar un robot y los sistemas operativos robóticos.

- Manejo básico de un robot industrial: la unidad de programación
- Movimiento del robot en modo manual
- Aspectos fundamentales para programar un robot
- Parámetros del sistema. Administrador de Archivos. Puesta en Marcha y Funcionamiento en Producción
- Programación de robots. Elementos básicos del lenguaje
- Instrucciones de posicionamiento: Tipos de instrucciones de movimiento.
- Prácticas: Realización de programas con el robot
- Instrucciones para la gestión de entradas y salidas
- Instrucciones para el diálogo con el operador.

Unidad didáctica III. Introducción a sistema operativo de robots (ROS) y aplicaciones

En la unidad I, el alumno adquiere conocimientos sobre los sistemas operativos robóticos. También se explican aplicaciones actuales y potenciales

- Identificación de Objetos

- Segmentación y reconocimiento
- Reconocimiento facial
- Reconocimiento de gestos
- Seguimiento de objetos
- Comprensión de movimiento
- Estructura de movimientos (SFM)
- Visión estéreo: percepción de profundidad mediante el uso de dos cámaras
- Movimientos
- Robots móviles
- Control
- Planificación
- Agarre de objetos.

Unidad didáctica IV. Historial de versiones

En la unidad II, el alumno conoce los modelos robóticos con aplicación de un sistema operativo.

- 2016 - Kinetic Kame
- 2015 - Jade Turtle
- 2014 - Indigo Igloo
- 2013 - Hydro Medusa
- 2012 - Groovy Galapagos
- 2012 - Fuerte
- 2011 - Electric Emys
- 2011 - Diamondback
- 2010 - C Turtle
- 2010 - Box Turtle
- 2010 - ROS 1.

Unidad de didáctica V. Robots y hardware relacionado.

En la unidad IV, el alumno debe conocer los robots y hardware soportado por ROS.

- Willow Garage PR2.
- Donaxi: robot de la UPAEP
- PR1: robot en Stanford.
- Robotnik: español.
- Robot de Shadow: mano robótica diestra motorizada (Europa).
- HERB: desarrollado por personal de Intel.
- STAIR I y II: robots de la Universidad de Stanford.
- Koen Buys, robot desarrollado en Bélgica.
- Aldebaran Nao: robot Humanoide desarrollado en la Universidad de Freiburg.

--

Criterios de desempeño

1. Participación activa en clase.
2. Ser puntuales.
3. Asistencia. Es muy importante. Tomar en cuenta el Reglamento Escolar.
4. Cumplir cabal y puntualmente con todas las actividades y trabajos.
5. Hacer los exámenes en las fechas programadas.
6. Trabajar en equipo.

Experiencias de Enseñanza / procesos y objetos de aprendizaje requeridos

1. Exposición del maestro de temas teóricos.
2. Exposición de alumnos de aplicaciones industriales.

Experiencias de aprendizaje.

1. Investigación de artículos en revistas de ciencia y tecnología.
2. Exposición de modelos de robótica OS

Recursos didácticos y tecnológicos (material de apoyo):

1. Laptop del instructor.
2. Cañón.
3. Pintarrón.
4. Conexión a internet.
5. Software Comercial.

<i>Bibliografía</i>	<i>Básica/ Complementaria</i>
Jason R. Briggs. (2012). Python for Kids: A Playful Introduction to Programming. 1 th edition. Edit. No Starch Pr.	<i>Básica</i>
Knowlton Jim (2009). Python. 1 ^{ra} edición. Edit. Anaya Multimedia.	<i>Básica</i>
Martelli Alex. (2007). Python. Guía de referencia. 1 ^{ra} edición. Edit. Anaya.	<i>Básica</i>
Kathy Ceceri. (2015). Making Simple Robots. Edit. Maker Media, Inc.	<i>Básica</i>

Massimo Bansi. (2016). Introducción al Arduino. Edit. Anaya Multimedia.	Básica
Quigley M. & Berger E. (2007). STAIR: Hardware and Software Architecture. 31/03/2017. de Robotics Workshop. Sitio web: http://stair.stanford.edu/	Básica
Quigley M, Gerkey B., Conley K., Faust J., Foote T., Leibs J., Berger E & Wheeler R.. (2010). ROS: an open-source Robot Operating System. 31/03/2017. de Computer Science Department, Stanford University. Sitio web: http://www.willowgarage.com/sites/default/files/icraoss09-ROS.pdf	Básica
Bass L., Clements, P., & Kazman, R. (2003). Software architecture in practice. Patparganj, Delhi: Edit. Pearson Education.	Complementario

IV. Evaluación Formativa de las Competencias

#	Tipo (C,H, A)	Evidencias a evaluar	Criterios de evaluación	Técnicas e Instrumentos de Evaluación	Ponderación %
1	C	Examen parcial	Se evaluará el nivel de conocimientos adquiridos en relación a la unidad I	Examen escrito	20 %
2	H, A	Exposiciones de programas para robótica	Se evaluará la capacidad, habilidades y actitudes en relación a trabajo en equipo, lectura y análisis de programas de robótica, exposición, organización de ideas.	Diseño, debate, organización y presentación de programas de robots	30 %

3	C	Examen parcial	Se evaluará el nivel de conocimientos adquiridos en relación a la unidad II	Examen escrito	20 %
4	C	Examen parcial	Se evaluará el nivel de conocimientos adquiridos en relación a las unidades III, IV y V	Examen escrito	20 %
5	H, A	Participación activa en clase	Se evaluarán las habilidades de comunicación, organización y actitudes de trabajo y compromiso del alumno	Participación en clases y asistencia	10 %
				Total	100 %

C: Conocimientos H: Habilidades A: Actitudes