Lok Jagruti Kendra University
University with a Difference

Diploma in Automation & Robotics

LJ Polytechnic

Course Code: 025120504

Robotics Programming

Programme / Branch Name			Diploma in Automation & Robotics			
Course Name	Robotics Programming				**Course Code**	025120504
Course Type	HSSC	BSC	ESC	PCC	OEC	PEC

Legends: HSSC: Humanities and Social Sciences Courses BSC: Basic Science Courses
ESC: Engineering Science Courses PCC: Program Core Courses
OEC: Open Elective Courses PEC: Program Elective Courses

1. Teaching and Evaluation Scheme

Teaching Hours / Week				**Evaluation Scheme**			
L	**T**	**P**	**Total Credit**	**CCE**	**SEE (Th)**	**SEE (Pr)**	**TOTAL**
3	0	2	4	50	50	50	150

Legends:
L: Lectures T: Tutorial P: Practical
CCE: Continuous & Comprehensive Evaluation
SEE (Th): Semester End Evaluation (Theory)
SEE (Pr): Semester End Evaluation (Practical)

2. Prerequisites

- ✓ Applied Physics
- ✓ Engineering Mathematics-I
- ✓ Engineering Mathematics-II
- ✓ Applied Mechanics
- ✓ Fundamentals of Electricals & Electronics Engineering
- ✓ Python Programming

3. Rationale

Robotics Programming at the Diploma Level can provide students with valuable interdisciplinary skills and knowledge. It requires a combination of mathematics, physics, electronics, and computer science, promoting critical thinking and problem-solving abilities. The growing use of automation and robotics in various industries is creating a demand for skilled professionals, making it a promising career path.

4. Objectives

- ✓ Introduce students to fundamental robotics concepts and programming, including sensors and actuators, kinematics, and dynamics.
- ✓ Develop students' programming skills, including conditionals, loops, and functions, and provide hands-on experience with microcontrollers or simulators.
- ✓ Teach path planning and obstacle avoidance for practical robotic applications.
- ✓ Promote critical thinking, problem-solving skills, and innovation in robotics programming, while considering ethical and social implications.
- ✓ Prepare students for successful careers in robotics programming and related fields.

5. Contents

Unit No.	Unit Name	Topics	Learning Outcome	% Weightage	Hours
1.	**Basics of Robotics**	1.1. Introduction 1.2. Robot Introduction 1.3. Seven Criteria of Defining a Robot 1.4. Robot Controllers 1.5. Major Components 1.6. Robot Vocabularies 1.7. Robotics Middleware Basics	• To understand the concept of robotics and its importance • To identify and define the basic components of a robot • To familiarize with different types of robot controllers and their uses	14	6
2.	**Ubuntu Linux For Robotics**	2.1. Introduction 2.2. GNU/Linux 2.3. Installing Ubuntu 2.4. Installing VirtualBox 2.5. Playing with the Ubuntu 2.6. Useful Ubuntu Applications 2.7. Shell Commands	• To understand the basics of GNU/Linux and its role in robotics • To install and configure Ubuntu for robotics development and experimentation • To acquire skills to perform basic tasks in Ubuntu and use it as an operating system for robotics development	14	6
3.	**Python For Robotics Programming**	3.1. Introduction to The Python Language 3.2. Python in Ubuntu Linux 3.3. Introduction to Python Interpreter 3.4. Installing Python on Ubuntu 16.04 LTS 3.5. Verifying Python Installation 3.6. Writing First Code 3.7. Understanding Python Basics	• To introduce Python programming language and its role in robotics • To understand the basics of Python programming, including data types, variables and control structures • To acquire skills to manipulate strings, lists, and other data structures in Python	24	10
4.	**ROS Fundamentals**	4.1. The ROS Equation 4.2. Robot Programming Before and After ROS Installing ROS 4.3. Robots and Sensors Supporting ROS 4.4. Popular ROS Computing Platforms	• To understand the importance of ROS in modern robotics • To familiarize with popular ROS computing platforms and robots and sensors that are compatible with ROS	24	10

		4.5. ROS Architecture and Concepts	• To apply the knowledge and skills acquired to build simple robotics applications using ROS		
5.	**Programming with ROS**	5.1. Programming Using ROS 5.2. Creating a ROS Workspace and Package 5.3. Using ROS Client Libraries 5.4. Programming Embedded Boards	• To understand how to program using ROS • To create a ROS workspace and package for organizing and managing ROS projects • To programme embedded boards such as Arduino and Raspberry Pi for use in ROS robotics applications	24	10
				Total Hours	**42**

6. List of Practicals / Exercises

The practicals/exercises have been properly designed and implemented in an attempt to develop different types of skills, so that students can acquire the competencies/programme outcomes. Following is the list of practicals/exercises.

Sr. No.	Practical / Exercises	Key Competency	Hours
1.	To demonstrate of various types of Robot Controllers	Demonstration skills, Knowledge of Robot Controllers	2
2.	To demonstrate of Major Components of Robots	Demonstration skills, Knowledge of Robot Components	2
3.	Installing Ubuntu	Technical skills, Knowledge of Ubuntu	4
4.	Installing VirtualBox	Technical skills, Knowledge of VirtualBox	4
5.	Installing Python on Ubuntu 16.04 LTS	Technical skills, Knowledge of Python and Ubuntu	4
6.	Hands on Programming with ROS	Programming skills, Knowledge of ROS	8
7.	Seminar Presentation	Presentation skills, Research skills, Knowledge of the topic	4
		Total Hours	**28**

7. Suggested Specification Table for Evaluation Scheme

Unit No.	Unit Name	Distribution of Topics According to Bloom's Taxonomy					
		R %	U %	Ap %	C %	E %	An %
1.	Basics of Robotics	30	40	20	5	5	-
2.	Ubuntu Linux For Robotics	10	40	30	10	5	5
3.	Python For Robotics Programming	20	40	20	10	5	5
4.	ROS Fundamentals	20	40	20	10	5	5
5.	Programming with ROS	10	30	30	20	5	5

Legends: R: Remembering U: Understanding
App: Applying C: Creating
E: Evaluating An: Analyzing

8. Textbooks

1) Lentin Joseph, Robot Operating System (ROS) for Absolute Beginners: Robotics Programming Made Easy, 1st Edition
2) Jonathan Cacace; Lentin Joseph, Mastering ROS for Robotics Programming: Design, build, and simulate complex robots using the Robot Operating System, 2nd Edition, Packt Publishing

9. Reference Books

1) Introduction to Robotics by Philippe J. Martin
2) Programming Robots with ROS: A Practical Introduction to the Robot Operating System by Morgan Quigley, Brian Gerkey and William D. Smart
3) Handbook of Robotics by B. Siciliano and O. Khatib (Editors)
4) Modern Robotics: Mechanics, Planning, and Control by Kevin M. Lynch and Frank C. Park
5) Hughes, C. and Hughes, T., Robot programming: a guide to controlling autonomous robots. Que Publishing
6) Quigley, M., Gerkey, B. and Smart, W.D., Programming Robots with ROS: a practical introduction to the Robot Operating System, O'Reilly Media, Inc.
7) Anil Mahtani, Luis Sanchez, Enrique Fernandez, Aaron Martinez, Lentin Joseph. ROS Programming: Building Powerful Robots. Packt Publishing

10. Open Sources (Website, Video, Movie)

1) ROS (Robot Operating System) - http://www.ros.org/
2) Gazebo - http://gazebosim.org/
3) OpenCV - https://opencv.org/
4) PyRobot - https://pyrobot.org/
5) RobotPy - https://robotpy.github.io/
6) TurtleBot - http://www.turtlebot.com/
7) https://onlinecourses.nptel.ac.in/noc19_me74/preview
8) https://onlinecourses.nptel.ac.in/noc20_de11/preview
9) https://onlinecourses.nptel.ac.in/noc20_me03/preview
10) http://www.osrobotics.org/osr/