3D Educational Services Inc

Introduction to Programming Concepts And Robotics with Raspberry Pi

Course Overview:

The following is the course information details for a semester long session to teach Programming concepts and Robotics using Raspberry Pi.

The course will be taught twice a week for 1 hour each for 16 weeks (semester). This document provides the lesson plans based on weekly schedule. The course also includes the following apart from the lesson plans:

- Vocabulary sheets
    - Programming concepts
    - Robotics concepts
    - Python Programming Language
    - MATLAB Simulink
- Assignments
- Quizzes
    - Programming concepts
    - Robotics lessons
    - Python Programming Language
    - MATLAB Simulink
- Appendices to include
    - Introduction to basic electronics
    - Troubleshooting tips and techniques
    - Links to reference material

Week 1: Session 1

Intro to Raspberry Pi **-** This lesson introduces Raspberry Pi as a computing platform and how to program Raspberry Pi using Python.

Activity:

- Introduction to Raspberry Pi 3 Model B hardware
- Download and create the latest Raspbian Operating System Image on SD card for Raspberry Pi
- Configure the wireless networking options to access Raspberry Pi

Lesson objectives:

- ➢ Understand the different features of Raspberry Pi
- ➢ Understand the basics of Raspbian Linux OS software and its flavors
- ➢ Understand different OS available for Raspberry Pi and their advantages and disadvantages

Week 1: Session 2

Intro to Raspberry Pi **-** This lesson will provide students all the basic tools and techniques to work with the Raspberry Pi.

Activity:

- Configure remote desktop and other software to access Raspberry Pi
- Students are able to boot and access their Raspberry Pi
- Install and access Python IDE on Raspberry Pi
- Install MATLAB Simulink for Raspberry Pi

Lesson objectives:

- ➢ Students are able to boot the Raspberry Pi on the network and access it remotely
- ➢ Install Python IDE and MatLab Simulink software on Raspberry Pi

Reference: Raspberry Pi Vocabulary Sheets

Week 2: Session 1

Raspbian Linux OS **-** This lesson introduces a conceptual framework for thinking of a computing device as something that uses code to process one or more inputs and send them to an output(s). This chapter introduces students to basics of computing and Raspbian Operating System (OS) based on Linux.

Activity:

a. Understand Raspbian OS (Linux) and how to connect to Raspberry Pi and navigate the Linux OS
b. Learn to use basic Linux commands and Python programming language

Lesson objectives:

- Understand the four components that make up a computer and their functions.
- Understand that the Raspberry Pi takes input, and after processing the input, produces output.
- Learn the different types of information Raspberry Pi takes in as input.
- Apply this knowledge by creating a Raspberry Pi program that takes input and produces an output.

Week 2: Session 2

Raspbian Linux **-** This lesson builds on the previous session of Raspbian Linux OS with reference to the Raspberry Pi environment and its features. In this session students will gain an understanding of Raspberry Pi's General Purpose Input / Output (GPIO) Pins.

Activity:

a. Understand GPIO (general purpose Input/Output) pins on Raspberry Pi
b. Create Programs to blink LEDs on Raspberry Pi and understand sleep function

Lesson objectives:

- Understand different GPIO pins that are available on Raspberry Pi and their functionality
- Learn how to safely use Raspberry Pi with different electronic components
- Learn to read/write data from GPIO pins and interpret them

Reference: Introduce the Raspbian Linux OS vocabulary sheet

Week 3: Session 1

Introduction to Variables **-** This lesson introduces the use of variables to store data or the results of mathematical operations. Students will practice giving variables unique and meaningful names. We will also introduce the basic mathematical operations for adding, subtracting, multiplying, and dividing variables.

Activity: Write a program to simulate -

a. "Flipping Coin" or "Voting Machine"
b. Hot or Cold – "hot chocolate" or cold "milk shake"

Lesson Objectives:

- ➢ Understand what variables are and why and when to use them in a program. Also understand how to manipulate variables in your program.
- ➢ Learn how to create a variable, set the variable to an initial value, and change the value of the variable within a Python program.
- ➢ Learn how to use the basic mathematical functions for adding, subtracting, multiplying, and dividing variables.
- ➢ Understand the application of "Boolean" variable; True or False
- ➢ Apply the above knowledge that uses variables in a program.
- ➢ Learn how to write pseudocode and flowchart

Reference: Introduce the Pseudocode & Flowchart Vocabulary Sheet

Extra Credit: Write a program that can work as a clock!

Week 3: Session 2

The idea of Conditionals **-** This lesson introduces the Logic blocks such as 'If…then' and 'If…then…else'. Students practice skills of creativity, problem-solving, and collaboration.

Activity: Write a program to simulate –

a. Use conditionals with random number (> or < 100) & show "number"
b. Rock, paper, scissors – Use Random number between 0 -2 to simulate game!

Lesson objectives:

- ➢ Understand what conditional statements are, and why and when to use them in a program.
- ➢ Learn how to use the Logic blocks 'If…then' and 'If…then…else'.
- ➢ Practice using the Logic blocks so different conditions yield specified outcomes.
- ➢ Demonstrate understanding and apply skill by collaborating with classmates to create a game that uses Raspberry Pi and a program that correctly and effectively uses conditionals.

Extra Credit: Write a program that can work as a clock!

Week 4: Session 1

Iteration & Looping **-** This lesson introduces the concept of looping and iteration. Presents the ‘While’ loops as a combination of an iteration and a conditional statement.

Activity: Write a program to simulate different loops: ‘While’ and ‘For’

1. Write message “Hello, Raspi!” in a loop forever!
2. Write a program that keeps on taking input from user till he/she enters ‘X’ at which time the program will terminate
3. Write a function to count odd numbers from 1 to 20. Can you write another program for even numbers?

Lesson objectives:

- ➢ Understand the value of iteration in programming
- ➢ Understand looping as a form of iteration
- ➢ Learn how and when to use the loops ‘while’, and ‘for’
- ➢ Apply the above knowledge that uses iteration and looping as an integral part of the program

Extra Credit:

Week 4: Session 2

Working with Booleans **-** This lesson introduces the use of the Boolean data type to control the flow of a program, keep track of state, and to include or exclude certain conditions.

Activity: Write program to simulate –

1. Double “Coin Flipper”
2. “Voting machine” for two candidates in the race

Lesson objectives:

- ➢ Understand what Booleans and Boolean operators are, and why and when to use them in a program.
- ➢ Learn how to create a Boolean, set the Boolean to an initial value, and change the value of the Boolean within a Raspberry Pi program.
- ➢ Learn how to use the Random function for coin flips or voting.
- ➢ Apply the above knowledge and skills to create a unique program that uses Booleans and Boolean operators as an integral part of the program.

Extra Credit: “Voting machine” for multiple candidates in the race!

Write a program to simulate vote counting for multiple candidates and displaying vote counts for each candidate.

Week 5: Session 1

Introduction to Sensors **–** In this lesson students will learn how to read sensor data using Raspberry Pi's GPIO pins and respond to that input. Raspberry Pi uses only digital input and hence only sensors with digital input can be used.

Sensors:

1. IR Sensor to detect black line
2. Ultrasonic sensor to detect obstacle
3. Light sensor to detect ambient light

Activity: Write programs

- ➢ To read analog and digital signal data from sensors
- ➢ To interpret data from different sensors and make decisions

Lesson objectives:

- ➢ Understand how various sensors can be used with Raspberry Pi
- ➢ Understand the difference between 3.3V and 5V input GPIO pins on Raspberry Pi

Week 5: Session 2

Arrays **-** This lesson introduces the fundamental concept of storing and retrieving data in an ordered fashion using Arrays or Lists.

Activity: Write program to simulate –

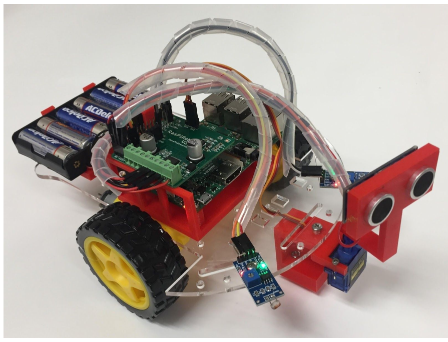
- ➢ Headband Charades – Create an array of words that can be used as part of a charades-type game.
- ➢ Capturing the sensor data over a period of time at a fixed interval and obtaining the average of the data.

Lesson objectives:

- ➢ Explain the steps they would take to sort a series of numbers.
- ➢ Recognize three common sorting algorithms.
- ➢ Practice creating Arrays.
- ➢ Practice storing and retrieving values in Arrays.
- ➢ Learn common Array operations such as setting and getting values by index.

Week 6: Session 1

Introduction to the Robotics - This lesson covers the introduction to robotics, what is robotics and applications of robotics. This lesson also introduces students to "RaspiRobot Board V3" – motor driver board to use along with Raspberry Pi 3 Model B and its features. In addition this lesson will introduce some basic electronics and components like sensors that will be used in this robot.



Activity:

- ➢ Identify various components of the robot
- ➢ Assemble the robot as per the instructions
- ➢ Attach external sensors – line follower, light sensor

Lesson objectives:

- ➢ Understand "what a robot is?" and different parts of the robot
- ➢ Build the Robot car and attach sensors

Week 6: Session 2

Programming the robot - This lesson covers the basic programming for the robot, students will program the robot to perform basic tasks like move forward, move backward, spin and go along a straight line (no sensors are used).

Activity: Write program to "perform" Robot movements

- ➢ Understand how to use the Robot to perform various tasks & how the RaspiRobot Board V3 works with Raspberry Pi 3 Model B
- ➢ Move Forward, Backward and spin – Create code to move robot forward, backward and spin
- ➢ Test different components of the robot – like motors and sensors by creating simple programs and understand the troubleshooting techniques

Lesson objectives:

- ➢ Understand how to program the robot using Python
- ➢ Understand how to use Thonny and linux vi editors on Raspberry Pi
- ➢ Understand how to control robot movements

Reference: Introduce Raspberry Pi Robot Vocabulary Sheet

Week 7: Session 1

Robot Navigation - This lesson covers the basics of robot navigation, the different techniques involved in navigating the robot and how students can program the robot.

Activity: Understand how to program the robot to navigate in specific ways

1. Program the robot to navigate in the path of a square. Try other shapes as well like number '8', pentagon, etc
2. Write pseudocode and validate with the rest of the class
3. Write flowchart for the program
4. Identify the programming constructs to be used with Python

Lesson objectives:

- ➢ Understand how robots navigate
- ➢ Program the robot to follow the path of a square and explore more complex navigation paths
- ➢ Write pseudocode for the program
- ➢ Write flowchart for the program

Week 7: Session 2

Robot Navigation with Encoders – In this lesson students will learn about encoders for wheels and how robots can be programmed to move precisely to move from one point to another using encoders.

Activity: Understand how to use encoders with robots

1. Assemble the encoders onto the robot to sense wheel rotation
2. Program the robot to use encoders and navigate to exact location
3. Program the robot to calculate distance traveled by the robot and the speed with which it traveled based on encoder data

Lesson objectives:

- ➢ Understand how encoders work
- ➢ Understand how to read and interpret encoder data
- ➢ Understand how to navigate robot based on encoder data

Week 8: Session 1

Work with IR sensors - Detect black line - In this lesson students will learn about the IR sensor and how to use it with Raspberry Pi to detect black and white (and grey).

Activity: Understand how to use IR sensor with the robot

1. Attach the IR sensor to the robot and ensure all the wiring connections are ok
2. Program the robot to detect black line while moving and stop at black line
3. Write pseudocode and flowchart for the program
4. Use the Python program to read Raspberry Pi GPIO pins to detect black and white

Lesson objectives:

- ➢ Understand the IR Sensor and and how it works
- ➢ Ideate and design the program to solve the detect black line challenge
- ➢ Write pseudocode for the program
- ➢ Write flowchart for the program

Week 8: Session 2

Work with IR sensor – Follow Black Line – In this lesson students learn how to program the Raspberry Pi based robot to follow a black line using infrared sensor.

Activity: Understand how to use infrared sensor to follow black line

1. Understand how to read data from IR sensor and interpret it.
2. Create pseudocode and flowchart to solve the challenge.
3. Write Python program to navigate the robot to follow a black line using data from IR sensor.

Lesson objectives:

- ➢ Understand how to IR sensor works and how to use the data to make the robot follow a line
- ➢ Understand how line follow algorithm works and its variations
- ➢ Explore how to optimize the line follow algorithm and how additional IR sensor might help

Week 9: Session 1

Work with Sensors - Ultrasound Obstacle Detection - This lesson covers the basics of what ultrasound sensors are and how they work. Students will experiment with ultrasound sensor and program the robot to detect obstacles.

Activity: Understand how to use ultrasound sensors to program the robot to detect obstacle

1. Define the challenge - Use ultrasound sensor to detect obstacles
2. Write pseudocode and validate with the rest of the class
3. Write flowchart for the program
4. Capture the sensor data and interpret the results, also work to optimize the obstacle detection by taking multiple readings and taking the average of the values

Lesson objectives:

- ➢ Understand what is ultrasound, ultrasound sensor and how it works
- ➢ Ideate and design the program to solve the obstacle detection challenge
- ➢ Write pseudocode for the program
- ➢ Write flowchart for the program

Week 9: Session 2

Work with Sensors (Ultrasound Obstacle Detection..continued) - This lesson builds on the previous session of Ultrasound sensors and its application as obstacle detector. In this session students will use the Ultrasound to follow the wall.

Activity: Understand how to use ultrasound sensor to follow along the edge of a wall

- ➢ Understand how to read distance data from ultrasound sensor
- ➢ Create pseudocode and flowchart to read and make decisions using two different ultrasound sensors
- ➢ Understand how to make decisions from reading data from multiple ultrasound sensors
- ➢ Program the robot to follow along the edge of a wall based on sensor data

Lesson objectives:

- ➢ Understand what is ultrasound waves are
- ➢ Understand what ultrasound sensors are and how they work
- ➢ Understand how to change behavior of robots based on data from multiple sensors

Week 10: Session 1

<u>Work with Sensors</u> (IR Collision Detection) – This lesson covers the basics of IR Collision detection sensor and how they interface with Raspberry Pi GPIO pins.

Activity: Work on using IR Collision detection sensor

1. Define the challenge - Use the IR collision detection sensor to detect obstacle
2. Write pseudocode and validate with the rest of the class
3. Write flowchart for the program
4. Create Python program to optimize the speed with which the robot can navigate while preventing collisions with obstacle

Lesson objectives:

- Understand what is an infrared collision detection sensor and how is it different than infrared line follow sensor
- Ideate and design the program to solve the collision detection challenge
- Write pseudocode for the program
- Write flowchart for the program

Week 10: Session 2

<u>Work with Photosensors</u> (Seek Light) – This lesson covers the basics of photocell light sensor and how they work. Students will learn to read data from two photocell sensors and use the data to navigate the robot towards light.

Activity: Work on using multiple photocell sensors

1. Define the challenge - Use the photocell light sensors to navigate the robot
2. Write pseudocode and validate with the rest of the class
3. Write flowchart for the program
4. Create Python program to read data from both the photosensors and decide which way the robot needs to turn

Lesson objectives:

- Understand what is photocell sensor and how it works
- Ideate and design the program to solve the 'seek light' challenge
- Write pseudocode for the program
- Write flowchart for the program

Week 11: Sessions 1 and 2

Accelerometer: This lesson will introduce the students to accelerometer sensor MPU-6050. Students will read data from the sensor and determine the movement of the robot.

Activity: Work on using accelerometer

1. Define the challenge - Use the accelerometer to determine whether the robot has bumped into something, use the accelerometer data to calculate speed of the robot
2. Write pseudocode and validate with the rest of the class
3. Write flowchart for the program
4. Create Python program to read data from accelerometer and determine the speed of the robot or stop when it bumps into something

Lesson objectives:

- Understand what an accelerometer is and how it works
- Ideate and design the program to solve the bump challenge and speed challenge
- Understand the complexity in reading sensor data and interpreting for decision making
- Write pseudocode for the program
- Write flowchart for the program

Week 12: Sessions 1 and 2

Gyroscope Sensor: This lesson will introduce the students to gyroscope sensor MPU-6050. Students will read data from the sensor and determine the rotation of the robot.

Activity: Work on using gyro sensors

1. Define the challenge - Use the gyro sensors to make the robot move
2. Write pseudocode and validate with the rest of the class
3. Write flowchart for the program
4. Create Python program to read data from the gyro sensor and decide the movement accordingly

Lesson objectives:

- Understand a gyro sensor and how it works
- Ideate and design the program to solve the gyro challenge
- Write pseudocode for the program
- Write flowchart for the program

Week 13: Sessions 1 and 2

Final Project - Dancing Robots – The objective of the final project is to apply all the different programming skills and robotic techniques that the student has learned during the course.

Session 1 Activity:

- Review the challenge and brainstorm with rest of the class to gain a better understanding of the challenge
- Teacher to review the rubrics with the students and help them on the weightage that is given to the different aspects of the challenge, for example…
    - Robot design and navigation techniques
    - Programming skills
    - Creativity

Session 2 Activity:

- Write Pseudocode and Flowchart for the challenge
- Understand how robots will work synchronously during the dance and work on the radio communication between robots
- Each student programs the robot to validate radio communication works

Week 14: Sessions 1 and 2

Final Project - Dancing Robots – The objective of the final project is to apply all the different programming skills and robotic techniques that the student has learned during the course.

Session 1 Activity:

- Students work to test their robots are working as per the programs
- Student teams will also test whether their robots are working synchronously to perform the dance
- Use different trigger mechanisms – Buttons & Radio on Microbit as remote.
- Students will troubleshoot and debug problems and fix all issues

Session 2 Activity:

- Student teams to perform final check before the final demo
- Student teams will demo their robots to perform the dance
- Judging and final closing comments from teachers

Weeks 15 and 16: Sessions 1 and 2

Assessments and Quizzes:

- Raspberry Pi and Linux Commands
- Python Programming Syntax
- Robotics definitions and its applications
- Sensors definitions and its applications