

COURSE CSE1240: ROBOTICS PROGRAMMING 1

Level: Introductory

Prerequisite: CSE1110: Structured Programming 1

Description: Students use an appropriate Robot Control Language (RCL) to design, develop, implement and debug robotics programs that employ standard structured programming constructs and simple data structures. In the process, they develop a general understanding of robots and the robotics environment.

Parameters: Access to appropriate computer equipment, software, support materials and the Internet. More specifically, students must have access to either the physical (real) or virtual (simulated) robotic environment they will require to design, write and debug RCL scripts or programs.

Supporting Courses: CSE1010: Computer Science 1
 CSE1120: Structured Programming 2
 ELT1130: Robotics 1
 ELT1140: Robotics Applications 1

Outcomes: The student will:

1. demonstrate a general understanding of robotics including the nature of robots, robotic control systems and the robotic environment

- 1.1 describe robots in general terms including:
 - 1.1.1 build a general mental model of robotic architecture by outlining the characteristics that are common to all robots
 - 1.1.2 describe the robotic characteristic of “agency”, the appearance of operating as autonomous agents and indicating the central role it plays in the concept of robot
 - 1.1.3 indicate how robots interact with their environment
- 1.2 describe robotic control systems in general terms including:
 - 1.2.1 describe the three robotic primitives associated with robotic behaviour and their role in robotic control
 - 1.2.2 describe the three main approaches to robotic control
 - 1.2.3 use the three robotic primitives to describe how each approach to robotic control is carried out
- 1.3 describe robotic environments in general terms including:
 - 1.3.1 describe the delimited nature of the robotic environment
 - 1.3.2 explain why robotic environments are circumscribed or limited in comparison with human environments
 - 1.3.3 describe the ideal relationship between a robot and its environment
 - 1.3.4 describe the issues associated with a mixed human/robot environment
- 1.4 compare and contrast virtual (simulated) robots and physical (real) robots in general terms including:
 - 1.4.1 compare and contrast their architecture
 - 1.4.2 compare and contrast their control systems
 - 1.4.3 compare and contrast their environment

2. **use a general understanding of robotics to analyze a robot operating within its environment**
 - 2.1 describe its architecture
 - 2.2 indicate how it displays “agency” or autonomous action
 - 2.3 categorize its control system
 - 2.4 describe its capabilities
 - 2.5 describe its relationship with its environment
 - 2.6 identify at least one task that the robot should be able to accomplish within its environment
 - 2.7 explain how either the robot and/or its environment could be modified to increase the number and type of tasks it could accomplish
3. **design a robotics system consisting of at least one robot, associated control systems and environment capable of carrying out a simple set of predetermined tasks**
 - 3.1 identify the general tasks the robot will be required to carry out by:
 - 3.1.1 breaking tasks into simpler tasks and continuing the process until the tasks are reduced to primitives
 - 3.1.2 establishing the task sequence and creating a representation of those tasks
 - 3.2 describe and illustrate the environment in which the robot will be required to operate by:
 - 3.2.1 identifying elements of the environment that the robot will be able to manipulate
 - 3.2.2 identifying manipulatable elements that will act as task resources and task obstacles
 - 3.2.3 identifying variations in the environment that the robot will be able to detect; e.g., light, colour, sound
 - 3.2.4 setting the outer limits of the environment
 - 3.2.5 determining the location and type of internal barriers in the environment
 - 3.2.6 incorporating safety elements into the environment that will protect both the humans and robots operating in the environment
 - 3.3 identify the capabilities the robot will require to carry out set tasks including:
 - 3.3.1 sensing requirements
 - 3.3.2 mobility requirements
 - 3.3.3 manipulation requirements
 - 3.3.4 power requirements
 - 3.3.5 processing requirements; e.g., both calculation and data storage
 - 3.4 determine the control approach to be used to:
 - 3.4.1 determine if the robot has the capacity for autonomous operation
 - 3.4.2 determine what level of operator control will be required if the robot is not fully autonomous
 - 3.5 design the robot using the tasks to be accomplished, proposed environment, required capabilities, and control approach as parameters to determine what:
 - 3.5.1 type of kinematic chain or body will be required to provide a platform for the other components
 - 3.5.2 actuators and end effectors will be required and how they will be mounted on the body
 - 3.5.3 sensors will be required and how they will be mounted on the body
 - 3.5.4 control components will be required and how they will be mounted on the body
 - 3.5.5 power or energy components will be required and how they will be mounted on the body
 - 3.6 check your design for congruency against the task list to be accomplished and with the proposed environmental specifications
 - 3.7 modify the design, as required
 - 3.8 carry out the design process sequentially using a top-down approach and employing stepwise refinement
4. **use an iterative process to build the environment, robot and controlling mechanism called for by the design**
 - 4.1 construct that portion of the environment required for the first task or tasks in the task sequence
 - 4.2 assemble as much of the robot, as is required, to accomplish the task or tasks

- 4.3 write algorithms that use structured programming approaches to accomplish the task or set of tasks including:
 - 4.3.1 use appropriate problem decomposition techniques to break the task into subtasks
 - 4.3.2 identify the sense, plan and action requirements of each subtask
 - 4.3.3 further decompose each subtask into smaller blocks, as required, using the appropriate structures to control program flow
 - 4.3.4 write the algorithm in an approved format such as a structured chart or pseudocode
 - 4.3.5 use appropriate techniques to determine if the algorithm will achieve the original intent
- 4.4 use an RCL capable of writing structured code to translate the algorithm for the set of tasks into a program including:
 - 4.4.1 maintain the structure of the algorithm
 - 4.4.2 use appropriate basic or primitive data types and variable and constant names, as required, to hold data
 - 4.4.3 use literals and input commands to accept data from sensors to provide data for processing
 - 4.4.4 use operators and basic processing idioms as required; e.g., accumulation, determination of maximum or minimum values
 - 4.4.5 use appropriate selection and iteration structure; e.g., conditional, iterative blocks
 - 4.4.6 use output commands to display processed data to the operator as well as activate actuators; e.g., motors, grippers
 - 4.4.7 document appropriately
- 4.5 load and execute the program tracking and eradicating errors by:
 - 4.5.1 testing each of the physical subsystems of the robot(s) to eliminate engineering errors
 - 4.5.2 testing the robot(s) within the appropriate section of the environment to confirm that the robot is interacting with the environment as called for by the algorithm
 - 4.5.3 using self-test code and check points, as well as observation, to eliminate run-time and internal logic errors
 - 4.5.4 comparing the robot's actions with the intent of the algorithm
 - 4.5.5 modifying the original task list, environment, algorithm and/or program, as required
- 4.6 participate in interim critiques throughout the iterative process; e.g., planning, analysis, design, testing, evaluation
- 5. demonstrate basic competencies**
 - 5.1 demonstrate fundamental skills to:
 - 5.1.1 communicate
 - 5.1.2 manage information
 - 5.1.3 use numbers
 - 5.1.4 think and solve problems
 - 5.2 demonstrate personal management skills to:
 - 5.2.1 demonstrate positive attitudes and behaviours
 - 5.2.2 be responsible
 - 5.2.3 be adaptable
 - 5.2.4 learn continuously
 - 5.2.5 work safely
 - 5.3 demonstrate teamwork skills to:
 - 5.3.1 work with others
 - 5.3.2 participate in projects and tasks
- 6. make personal connections to the cluster content and processes to inform possible pathway choices**
 - 6.1 complete/update a personal inventory; e.g., interests, values, beliefs, resources, prior learning and experiences
 - 6.2 create a connection between a personal inventory and occupational choices