

## Assignment #2

Total score: 50

Due date: 2/27 (Monday)

---

In this assignment, we will learn about **Robot Operating System (ROS)**, its applications, and potential for making and programming your own robot.

**ROS** is a collection of software frameworks, packages, and libraries for robot software development. ROS was originally developed in 2007 by the Stanford AI lab with the goal of providing a single robot platform that integrates various methods from all areas of AI, including vision, navigation, planning, machine learning, reasoning, and natural language processing. The project was started at a robotics research lab, developed by Willow Garage, and transitioned to the Open Source Robotics Foundation in 2013. Since then, like other open source distributions such as Linux there has been several different distributions/releases (versioned set of ROS packages) including Box Turtle (2010), Indigo Igloo (2014), Jade Turtle (2015), Kinetic Kame (2016), etc.

ROS contains many open source implementations of common robotics functionalities and algorithms which are organized into packages. Like an operating system (OS), ROS provides standard operating system services and libraries as hardware abstraction, device control, commonly used functionality, message passing between processes, and package management.

Software in the ROS Ecosystem can be separated into three groups including language- and platform-independent infrastructure and tools used for building and distributing ROS-based software, ROS client library implementations such as **roscpp** (C++), **rospy** (Python), and **roslisp** (Lisp), and application-related packages.

Some **useful sources** for ROS can be found at:

- [wiki.ros.org](http://wiki.ros.org)
- A gentle introduction to ROS by Jason M. O’Kane (<https://cse.sc.edu/~jokane/agitr/agitr-letter.pdf>)
- Introduction to ROS programming (<http://www.cs.utexas.edu/~todd/cs378/slides/Week8a.pdf>)
- ROS Education Portal (<http://edu.gaitech.hk>)
- Learn TurtleBot and ROS (<http://learn.turtlebot.com>)

### Required tasks in this assignment

(a) **Check** the following **prerequisites** to begin with ROS programming and complete this and next assignments:

- **Ubuntu Linux 14.04 LTS** (strongly recommended for Swarmathon-ROS)
- A reasonably fast computer
- Familiarity with software development in a Linux environment using commands, Make tool, Github, etc.
- Knowledge of programming C++/Python (C++ strongly recommended)

(b) **Install ROS Indigo** (Desktop-Full) from <http://wiki.ros.org/indigo/Installation/Ubuntu>.

(c) **Read and finish the Core ROS Tutorials for beginner level** to get familiar with ROS and ROS programming. Make sure you fully understand ROS environment, ROS file system, ROS package, nodes, topics, publisher and subscriber, and service. The basic knowledge about ROS and ROS programming may be evaluated through exams.

- (d) **Write C++ program** that makes the **Box Turtle** (NOT Turtlebot) autonomously navigate on a rectangle loop as shown below. The Turtle can start on any location and follow a rectangular shape of path. Once it comes back to the original location, the navigation is stopped. Useful lectures on robot navigation can be also found at [edu.gaitech.hk](http://edu.gaitech.hk).



- (e) **Take a snapshot** of your program result to put in your report along with your source code.

**Note:** Cal State Fullerton (CSUF) will participate in the **2017 NASA Swarmathon Competition** (see <http://nasaswarmathon.com>) which is a challenge to develop cooperative robotics (involving multiple robots). The event will occur April 18~20 at NASA Kennedy Space Center. The event is organized by the University of New Mexico. There will be two types of competition, virtual and physical competition. CSUF team will participate in the virtual competition this year and plans to participate in physical competition next year.

If your team is interested in participating in this event representing CSUF, please let me know ASAP. Your team may have to follow additional instructions to compete in this challenge, regardless of the course requirements.

- (f) **Write a brief report in Word format** including (a) team name, *member name(s)* and **email addresses** of all members; the *percentage contribution* to this assignment for each member. If a team cannot reach a consensus on individual contribution, describe the individual's claimed percent contribution with a brief description on specific tasks performed; (b) a brief description on the **strategy** (or **heuristic**) used to accomplish your goal and a **pseudo code** for your strategy (NOT source code); (c) a **snapshot** of your program result; The source code you wrote should be included as a separate file, NOT in this report; (d) optionally provide reference(s) to the source of the program specifying the URL, author when some portion or all of the program was reused (or copied) from other people's code.

**Note:** (a) ~ (c) are required. Item (d) is optional but it is required if you reused someone's source code.

**Warning:** Although the code reuse is allowed for this assignment, copying the code from other person or team in this class is strictly prohibited. Any one or team violating this rule will receive **ZERO** score for this assignment.

### How to submit this assignment

**Include** (a) your report, (b) ONLY the file(s) you created/modified (**DO NOT** include other ROS files you didn't change/add) in **ONE zipped file** by **your team name** and submit the **zipped file** to **Titanium**. For example, if your team name is "ABC", then the zip file name should be **ABC.zip**. **Only ONE** of your team members needs to submit your team's work.

### Grading criteria

- Grade will be based on the quality of your team's work based on the requirements and the written report.