

Модуль 02 - Robot Operating System

Написание простого издателя и подписчика

Резюме: Вы изучите термины и некоторые утилиты командной строки. Напишите узлы для простой связи между программами внутри робота.

Глава I

Преамбула

Система сообщений это всегда непростая задача в планировании приложений. Сегодня вы посмотрите, каким образом программы общаются между собой, используя сервис распределения данных (DDS).

Глава II

Инструкции

- Используйте эту страницу как единственное описание задач. Не слушайте никаких слухов и домыслов о том, как приготовить свой программное решение.
- Здесь и далее мы используем ROS2 Humble и C++/Python.
- Обратите внимание на тип ваших файлов и каталогов.
- Для оценки ваше решение должно находиться в вашем репозитории GitHub.
- Вы не должны оставлять никаких дополнительных файлов в своем каталоге, кроме тех, которые явно указаны в теме. Рекомендуется изменить ваш .gitignore, чтобы избежать конфликтных случаев.
- Когда вам нужно получить точный результат в вашей программе, запрещается отображать предварительно рассчитанный результат вместо правильного выполнения упражнения.
- Есть вопрос? Спросите у соседа справа. В противном случае спросите вашего соседа слева.
- Ваш справочник: коллеги/интернет/google.
- Вы можете задавать вопросы в telegram.
- Внимательно прочитайте примеры. Они вполне могут прояснить детали, которые явно не упомянуты в теме.

Глава III

Упражнение 01

Упражнение 01: Установка и настройка среды ROS

Каталог для хранения вашего решения: ex01/

Файлы которые должны находиться в каталоге: ex01.txt, .bashrc

Разрешенные функции:

Комментарии:

В этом упражнении вам нужно будет установить и настроить среду ROS.

Прочтите и следуйте этим руководствам шаг за шагом:

<https://docs.ros.org/en/humble/Tutorials/Beginner-CLI-Tools/Configuring-ROS2-Environment.html>

<https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Creating-A-Workspace/Creating-A-Workspace.html>

Файл .bashrc должен содержать следующие строки в конце файла:

```
# Set ROS2 Humble
source /opt/ros/humble/setup.bash
source ~/ros2_ws/install/local_setup.bash
```

После этого вам нужно выполнить команду “source ~/.bashrc” или открыть новое окно терминала.

После установки переменной окружения выполните команду:

```
printenv | grep ROS > ex01.txt
```

Затем сохраните файлы ex01.txt и ~/.bashrc в папку ex01.

Упражнение 02

Упражнение 02: Навигация по файловой системе ROS

Каталог для хранения вашего решения: ex02/

Файлы которые должны находиться в каталоге: package_path_rostopic.txt,
list_files_rospy_tutorials.txt

Разрешенные функции:

Комментарии:

В этом упражнении вам нужно будет посмотреть возможности команды `ros2 pkg` и получить следующие данные:

`package_path_ros2topic.txt` - должен содержать путь к ROS пакету «`ros2topic`».

`list_exec_action_tutorials.txt` - должен содержать список выполняемых файлов в каталоге пакета «`action_tutorials_py`».

Затем сохраните файлы `package_path_ros2topic.txt` и `list_exec_action_tutorials.txt` в папку `ex02`.

Упражнение 03

Упражнение 03. Создание пакета ROS

Каталог для хранения вашего решения: ex03/

Файлы, которые должны находиться в каталоге: созданный вами пакет ROS

Разрешенные функции:

Комментарии:

В этом задании вам нужно будет изучить tutorial «Создание пакета ROS».

Прочтите и следуйте этим руководствам шаг за шагом:

<https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Colcon-Tutorial.html>

<https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Creating-Your-First-ROS2-Package.html>

Создайте пакет ROS, который будет содержать только ваш логин в имени и зависеть от пакетов: std_msgs, rclpy и rclcpp.

Измените maintainer email на свой адрес электронной почты в файле package.xml.

Сохраните созданный пакет в папку ex03.

Упражнение 04

Упражнение 04 :Сборка пакета ROS

Каталог для хранения вашего решения: ex04/

Файлы должны находиться в каталоге: colcon_build.txt

Разрешенные функции:

Комментарии:

Используя документацию из предыдущего раздела, запишите выходные данные сборки вашего пакета colcon build в файл colcon_build.txt. Посмотрите, каким образом можно сохранить информацию из буфера stdout в Linux.

Сохраните файл colcon_build.txt в папку ex04.

Упражнение 05

Упражнение 05 : Понимание узлов ROS

Каталог для хранения вашего решения: ex05/

Файлы в каталоге: rosnode_list.txt, rosnode_info.txt

Разрешенные функции:

Комментарии:

В этом упражнении вам нужно будет изучить tutorial «Понимание узлов ROS».

Прочтите и следуйте этому руководству шаг за шагом:

<https://docs.ros.org/en/humble/Tutorials/Beginner-CLI-Tools/Understanding-ROS2-Nodes/Understanding-ROS2-Nodes.html>

Запустите узел TurtleSim, изменив его имя с помощью Remapping Argument на [ваш_логин]_turtle.

Сохраните вывод команды `ros2 node list` в файл `rosnode_list.txt`.

Сохраните вывод команды `ros2 node info` вашей ноды в файл `rosnode_info.txt`.

Сохраните файлы `rosnode_list.txt` и `rosnode_info.txt` в папку ex05.

Упражнение 06

Упражнение 06 : Понимание топиков ROS

Каталог для хранения вашего решения: ex06/

Файлы которые должны находиться в каталоге: cmd_vel.txt, eight.png

Разрешенные функции:

Комментарии:

В этом упражнении вам нужно будет изучить tutorial «Понимание топиков ROS».

Прочтите и следуйте этому руководству шаг за шагом:

<https://docs.ros.org/en/humble/Tutorials/Beginner-CLI-Tools/Understanding-ROS2-Topics/Understanding-ROS2-Topics.html>

Запустите TurtleSim. Запустите `ros2 topic echo /turtle1/cmd_vel` и сохраняйте данные в `cmd_vel.txt` на подобие упражнения 04. Используя только последовательность команд `ros2 topic pub`, нарисуйте цифру “восемь”. Сохраните скриншот результат turtlesim в файл `eight.png`.

Сохраните `cmd_vel.txt` и `eight.png` в папку `ex06`.

Упражнение 07

Упражнение 07 : Общие сведения о службах и параметрах ROS

Каталог для хранения вашего решения: ex07

Файлы которые должны находиться в директории : screenshot.png, rosservice_list.txt, parameter_server.txt

Разрешенные функции:

Комментарии:

В этом упражнении вам нужно будет изучить tutorial «Понимание служб и параметров ROS».

Прочтите и следуйте этим руководствам шаг за шагом:

<https://docs.ros.org/en/humble/Tutorials/Beginner-CLI-Tools/Understanding-ROS2-Services/Understanding-ROS2-Services.html>

<https://docs.ros.org/en/humble/Tutorials/Beginner-CLI-Tools/Understanding-ROS2-Parameters/Understanding-ROS2-Parameters.html>

Запустите TurtleSim и добавьте четырех черепашек с именами Leonardo, Raphael, Donatello, Michelangelo.

Установите для параметра /turtlesim/background_g значение 124.

Сохраните скриншот окна симулятора в виде файла screenshot.png.

Сохраните вывод команды `ros2 service list` в файл `rosservice_list.txt`.

Сохраните выходные данные всего сервера параметров в файле `parameter_server.txt`.

Сохраните `screenshot.png`, `rosservice_list.txt`, `parameter_server.txt` в папку `ex07`.

Упражнение 08

Упражнение 08 : Использование rqt_console и roslaunch

Каталог для хранения вашего решения: ex08

Файлы которые должны находиться в каталоге: rqt_console.csv, three_turtles.launch

Разрешенные функции:

Комментарии:

В этом упражнении вам нужно будет изучить tutorial «Использование rqt_console и ros2 launch».

Прочтите и следуйте этим руководствам шаг за шагом:

<https://docs.ros.org/en/humble/Tutorials/Beginner-CLI-Tools/Using-Rqt-Console/Using-Rqt-Console.html>

<https://docs.ros.org/en/humble/Tutorials/Intermediate/Launch/Creating-Launch-Files.html>

Создайте такой launch файл чтобы одновременно запускалось три **окна** симулятора черепах и вторая черепаха следовала за первой, а третья за второй. **Копирование поведения должно осуществляться через узлы mimic.**

Сохраните получившийся launch файл как three_turtles.launch

Запустите rqt_console. Установите уровень логирования на Warn.

После запустите ваши три черепахи.

Управляйте некоторое время первой черепахой с клавиатуры, хотя бы один раз столкнитесь со стеной. После чего сохраните вывод rqt_console в файл rqt_console.txt используя кнопку Save в интерфейсе.

Сохраните rqt_console.csv, three_turtles.launch в папку ex08.

Упражнение 09

Упражнение 09 : Создание ROS msg и srv

Каталог для хранения вашего решения: ex09

Файлы, которые должны находиться в каталоге: все необходимые файлы для компиляции вашего пакета описания сообщения + скриншот.

Разрешенные функции:

Комментарии:

В этом упражнении вам нужно будет изучить tutorиалы «Использование colcon edit для редактирования файлов в ROS» и «Создание ROS msg и srv».

Прочтите и следуйте этим руководствам шаг за шагом:

<https://github.com/colcon/colcon-ed>

<https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Custom-ROS2-Interfaces.html>

Создайте пакет содержащий описание нового типа сообщения “full_name_message” состоящего из трех строк: “last_name”, “name”, “first_name”.

Добавьте в пакет описание сервиса с именем “FullNameSumService” получающего три строки “last_name”, “name”, “first_name” и возвращающий обратно строку “full_name”.

Подключите модуль colcon edit и попробуйте отредактировать файл

FullNameSumService в вашем пакете. Сделайте скриншот открытого файла.

В папку ex9 скопируйте все необходимые файлы для компиляции вашего пакета и скриншот.

Упражнение 10

Упражнение 10 : Написание и проверка простого издателя и подписчика

Каталог для хранения вашего решения: ex10

Файлы, которые должны находиться в каталоге: все файлы, необходимые для компиляции и запуска вашего узла "text_to_cmd_vel".

Разрешенные функции:

Комментарии:

На выбор вам представлены варианты для C++ и Python:

<https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Writing-A-Simple-Cpp-Publisher-And-Subscriber.html>

<https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Writing-A-Simple-Py-Publisher-And-Subscriber.html>

Напишите узел с названием "text_to_cmd_vel" который будет подписываться на топик "cmd_text" в который можно посылать текстовые команды: "turn_right", "turn_left", "move_forward", "move_backward". Узел должен переводить их в команды скорости типа сообщения geometry_msgs/Twist, например 1 метр в секунду или 1.5 радиан в секунду и публиковать в топик /turtle1/cmd_vel так чтобы черепаха в turtlesim двигалась в соответствии с командами.

В папку ex10 скопируйте все необходимые файлы для компиляции и запуска вашего узла "text_to_cmd_vel".