

Подготовка к первой лабораторной работе

1. Установка Ubuntu (+VirtualBox если поверх windows)
2. Установка ROS

- a. устанавливаем версию -desktop-full
(<http://wiki.ros.org/noetic/Installation/Ubuntu>)
- b. запуск окружения ROS

Чтобы в терминале можно было работать с ROS, нужно его запускать каждый раз...

или добавить `source /opt/ros/noetic/setup.bash` в конфигурационный файл `.bashrc`

команда для этого

```
echo "source /opt/ros/noetic/setup.bash" >> ~/.bashrc  
source ~/.bashrc
```

3. Установка IDE

Чтобы установить редактор VS Code выполните `sudo snap install code --classic`

В VS code имеются полезные расширения, которые облегчат работу (их желательно установить): Python, Cmake, ROS.

установить roscore можно командой `sudo apt install python3-roslaunch` (Если вдруг его нет)

`printenv | grep ROS` - выводит все важные параметры из env

Откройте файл `~/.bashrc` в текстовом редакторе и проверьте содержимое.

4. Запустим ROS Master

Для это воспользуемся командой `roscore`

5. В новом терминале (не закрываем предыдущий) попробуем запустить что-то, для этого воспользуемся уже установленным пакетом `rospy_tutorials`

```
roslaunch rospy_tutorials talker
```

 (roslaunch запускает ноду из пакета)

Мы должны увидеть, как talker публикует сообщения с содержанием Hello world

6. Откроем еще один терминал и запустим в нем другую ноду из того же пакета

```
roslaunch rospy_tutorials listener
```

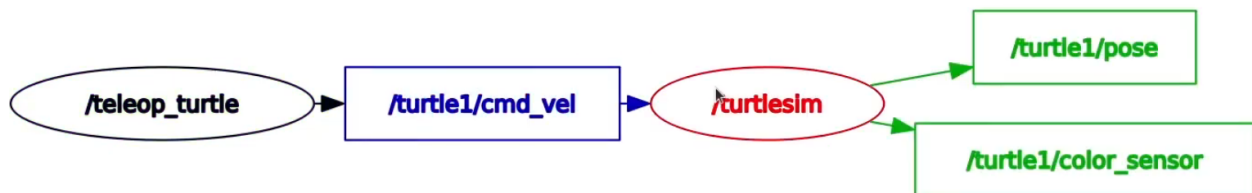
7. Воспользуемся утилитой `rqt_graph`, которая показывает вычислительный граф.

8. Запустим новую ноду из другого пакета

```
roslaunch turtlesim turtlesim_node
```

Снова запустим `rqt_graph` и проверим вычислительный граф, что мы видим?

9. В новом терминале запустим `roslaunch turtlesim turtle_teleop_key` и снова посмотрим на граф.



В итоге можно увидеть, что узел `/teleop_turtle` публикует сообщения в топик `/turtle1/cmd_vel` а который подписан `/turtlesim` и.д.

Чтобы создать свой пакет, нам понадобится создать свое окружение и воспользоваться catkin

Установка catkin выполняется через `apt-get install python3-catkin-tools`



Важно!!! Мы будем пользоваться **catkin build** вместо `catkin_make` не смотря на то, что во всех туториалах предлагается `catkin_make` !!!

Обязательным требованием является наличие `src` папки внутри рабочего пространства! Если вы один раз использовали `catkin_make` то в дальнейшем нельзя будет просто воспользоваться **catkin build**!

Пример создания окружения в папке catkin_ws

```
mkdir catkin_ws  
cd catkin_ws  
mkdir src  
catkin init  
catkin build
```

Для справок по catkin: <https://catkin-tools.readthedocs.io/en/latest/index.html>

Исследование пакета

Задание 1: найти расположение пакета `rospy_tutorials`

Задание 2: найти список всех установленных пакетов ROS

Задание 3: что вы можете сказать про пакет

<https://gitlab.com/beerlab/iprofi2023/demo/bachelor>