**Краткие методические рекомендации преподавателям по
дисциплине**

**«Инфраструктура разработки программного обеспечения для
мобильных роботов»**

Разделы дисциплины «Инфраструктура разработки программного обеспечения для мобильных роботов»

Всего 8 тем лекций:

- 1) Distribution data services (DDS)
- 2) Введение в ROS
- 3) Обмен сообщениями в ROS
- 4) Особенности реализации ПО с использованием ROS
- 5) Создание протоколов обмена для узлов робототехнической системы
- 6) Встроенные пакеты в ROS, упрощающие разработку для мобильных роботов
- 7) RTI Connex DDS
- 8) Open Splice

5 практических занятий

2 коллоквиума

Методика расчета оценки текущего контроля и итоговой оценки

К экзамену допускаются студенты, посетившие не менее 80% лекций и практических занятий и получившие за участие в 2 коллоквиумах оценки не менее «удовлетворительно».

Коллоквиумы проходят в устной форме и оцениваются по четырехбалльной шкале:

оценка «отлично» - на заданные вопросы даны исчерпывающие ответы,

оценка «хорошо» - вопросы раскрыты не полностью,

оценка «удовлетворительно» - ответы в принципе правильны, но в формулировках имеются существенные ошибки,

оценка «неудовлетворительно» - отсутствуют ответы на вопросы или содержание ответа не совпадает с поставленным вопросом.

Экзамен проводится по билетам. Критерии оценивания приведены в п. 6.1 Рабочей программы дисциплины.

Литература

Основная литература

1) Программирование в Robot Operating System [Текст] : учеб.метод. пособие / [К. В. Кринкин [и др.], 2018. 42, [1] с.

Дополнительная литература

1) Джозеф Л. Изучение робототехники с помощью Python [Электронный ресурс], 2019. 250 с

2) Каталог спецификаций OMG <https://www.omg.org/spec/#DDS>

Проверка базовых знаний

- Существующие подходы для программирования мобильных роботов.
- Архитектура publisherssubscriber.
- Фреймворки, обеспечивающие обмен сообщениями.
- Различные DDS решения. RTI Connex DDS, OpenSplice, ROS, ROS2.
- Связь высокоуровневого и низкоуровневого программирования.
- Способы эмуляции робота. Области применения ROS.
- Возможности ROS. Показательные примеры разработанных пакетов ROS.
- Turtle_bot, Turtle_sim. Исполняемые модули (Node).
- Эфир для приёма и передачи сообщений (Topic).
- Граф, связывающий Node и Topic.
- Расширение модели для переменного количества узлов при сохранении свойства realtime.
- «Нестандартное» управление Turtle_bot с использованием консоли
- Типы сообщений, содержащихся в Topic. Ограничение буфера обмена.
- Использование стандартных типов сообщений.
- Реализация собственных типов сообщений. Широковещательные сообщения. Сообщения, подразумевающие ответ. Блокирующие и неблокирующие сообщения в ROS.
- RQT_Graph как способ демонстрации текущего состояния системы. Rviz как способ визуализации.