

انجام این تمرین به صورت انفرادی می باشد

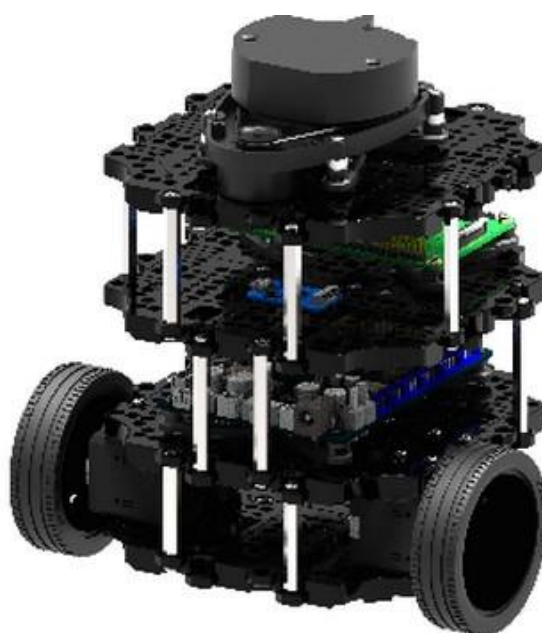
هدف از انجام این تمرین

هدف این تمرین آشنایی با مفاهیم پایه ROS است. برای انجام بخش پیاده سازی این تمرین، در ابتدا ویدیو ارائه شده برای نصب ROS و آموزش مفاهیم پایه ای آن را در مشاهده کنید. ROS یا سیستم عامل ربات، چارچوب نرم افزاری برای توسعه نرم افزار ربات است. ROS کتابخانه و ابزارهایی برای توسعه دهندگان نرم افزار، جهت ساخت کاربردهای نرم افزاری رباتیک فراهم می کند. این سیستم لایه انتزاعی سخت افزاری، راه اندازهای دستگاهها، کتابخانهها، ابزارهای بصری، ارسال پیام ها، مدیریت بسته ها و... را در اختیار کاربران قرار می دهد. لطفا نسخه ی مطرح شده در هندزان ترم جاری را مطابق با دستورالعمل ارائه شده در ویدیو نصب بفرمایید.

هدف از این تمرین تنها آشنایی با برنامه نویسی سمت سرور ROS می‌باشد. در این تمرین می‌خواهیم با استفاده از یک سناریو ساده با برنامه نویسی سمت سرور، ایجاد node و topic و همچنین publish/subscribe کردن پیام‌ها به وسیله topic آشنا شویم.

❖ شرح سناریو

موبایل رباتی مشابه شکل 1 در اختیار داریم. این ربات قابلیت چرخش درجا (rotate) دارد و همچنین سنسوری در بالای ربات قرار گرفته است که فاصله بر حسب سانتیمتر را از 4 جهت روبرو، پشت، راست و چپ ربات در اختیار ما قرار می‌دهد. هدف این است که با استفاده از دو قابلیت ذکر شده ربات (چرخش درجا و فاصله سنجی از طرفین) و به کمک ابزارهایی که ROS در اختیار ما قرار می‌دهد برنامه‌ای طراحی کنیم که ربات همواره به جهتی بچرخد که نزدیک‌ترین مانع به آن دقیقاً در پشتش قرار بگیرد.



شکل 1 turtlebot-burger

همانطور که در معیارهای برنامه نویسی تمیز (clean-code) گفته شده است که هر تابع یا method باید دقیقاً یک وظیفه مشخص انجام دهند در ROS هم هر گره یا node که ایجاد می‌کنیم باید دقیقاً یک وظیفه مشخص داشته باشد. از این رو برای طراحی سیستمی با مشخصات گفته شده کل سیستم را به 3 بخش اصلی تقسیم کرده و برای هر کدام از این بخش‌ها یک node ایجاد می‌کنیم. این سه بخش اصلی عبارت هستند از 1) سنسور تشخیص فاصله 2) کنترلر 3) موتور ها.

وظیفه هر کدام از 3 node نام برده شده در زیر قرار داده شده اند همچنین می‌توانید گراف مربوط را در شکل 2 مشاهده نمایید.

1. سنسور تشخیص فاصله: در این node در هر iteration چهار عدد رندوم تولید می‌شوند. این اعداد فاصله ربات از موانع موجود در چهار طرف ربات می‌باشند. این node باید مقادیر فاصله را در قالب یک custom-message درون topic مخصوص خود با نام distance قرار دهد. دقت شود که رنج دید سنسور فاصله 10 سانتیمتر تا 200 سانتی متر می‌باشد.

2. کنترلر: وظیفه این گره تشخیص بهترین جهت چرخش و کنترل چرخ های ربات با استفاده از مقادیر فاصله می‌باشد. برای این کار این گره ابتدا باید تاپیک distance را subscribe کند تا به مقادیر فاصله دسترسی داشته باشد. سپس اگر نزدیک ترین مانع در پشت ربات قرار نداشت باید بهترین جهت چرخش را تشخیص دهد. فرض کنید نزدیک ترین مانع در سمت چپ ربات قرار دارد. در این صورت اگر ربات ساعت گرد بچرخد بعد از 90 درجه و اگر پاد ساعت گرد بچرخد بعد از 270 درجه چرخش در جایگاه مطلوب قرار می‌گیرد. پس باید در این صورت کنترلر تشخیص دهد که چرخش مناسب 90 درجه و ساعت گرد است. در نهایت کنترلر باید پیام مناسب که شامل مقدار چرخش بر حسب درجه و جهت چرخش هست را در قالب custom-message در topic های مناسب مربوط به هر موتور قرار دهد.

نکته اول: ربات دارای 2 عدد موتور dc متصل چرخ، مانند شکل 1 می‌باشد.

نکته دوم: با توجه به شکل 1 اگر ربات بخواهد در جای خود حرکت چرخشی انجام دهد بدیتهتا چرخ ها در جهت مخالف همدیگر باید بچرخند. بنابراین مقادیر موجود در topic موتور 1 و 2 با هم متفاوت هستند.

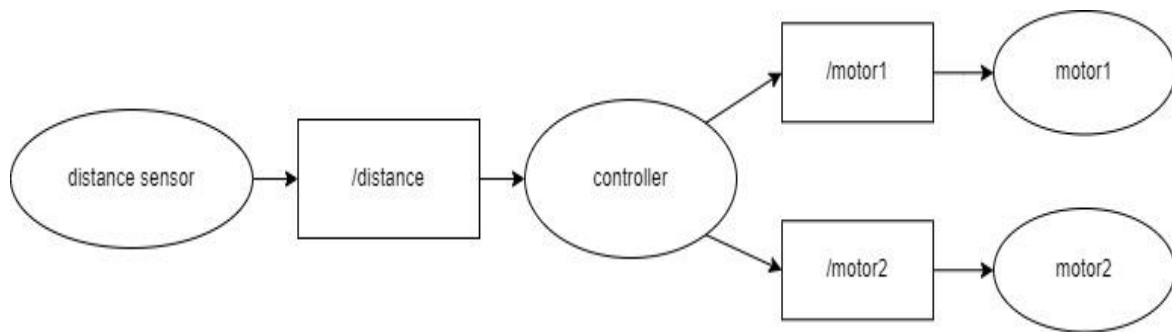
نکته سوم: این node هم subscriber است و هم publisher. شکل 2

3. موتور: این node تنها از topic مربوط به خود subscribe می‌کند و با استفاده از loginfo مقادیر دریافتی را درون ترمینال چاپ می‌کند.

در آخر برای اطمینان از درستی کارکرد برنامه و نمایش گراف نهایی می‌توانید دستور زیر را اجرا کنید:

```
rqt_graph
```

در انتها خروجی دستور بالا باید مشابه با تصویر زیر باشد.



شکل 1) گراف node و topic های سناریو اول

نکته آخر: برای راحتی کار فرض می‌کنیم که چرخش ربات هیچ زمانی نمی‌گیرد. به محض اینکه node کنترلر تصمیم می‌گیرد که ربات به چه اندازه و به چه جهتی بچرخد، چرخش صورت می‌گیرد. همچنین جهت قرارگیری یا state ربات باید همواره در node کنترلر update و ذخیره شود. جهت قرارگیری اولیه ربات دلبخواهی است.

❖ نحوه تحویل

کل فولدر پکیج مربوط به این تمرین را به همراه فایل pdf گزارش کار zip کرده و درون سامانه بارگزاری نمایید. در فایل گزارش کار باید صحت انجام درست و کامل سیستم به خوبی با عکس و اسکرین شات از ترمینال هایی که در آن node های مختلف را run کردید مشخص شده باشد. همچنین عکس گراف نهایی سیستم (خروجی rqt_graph) را در فایل گزارش کار قرار دهید

نکات تکمیلی در باب تحویل تمرین

1. تحویل گزارش در یک فایل pdf مطابق با قالب قرار گرفته در سایت کورسز و با نام‌گذاری HW1_StudentNumber می‌بایست تحویل داده شود.
2. فرمت فایل زیپ نیز به صورت HW1_StudentNumber باشد.
3. افراد می‌بایست تمرین را به صورت انفرادی انجام دهند.
4. دستیاران آموزشی ملزم به اجرا کردن کدهای شما نیستند. بنابراین هرگونه نتیجه و یا تحلیلی که در شرح سوال از شما خواسته شده است را به طور واضح و کامل در گزارش بیاورید. در صورت عدم رعایت این مورد، بدیهی است که از نمره تمرین کسر می‌شود.
5. برای انجام تمرین و پروژه تنها زبان برنامه‌نویسی مجاز پایتون می‌باشد.
6. تحویل کد به همراه تمرین لازم است و در صورت تحویل ندادن کد و اکتفا به گزارش، نمره‌ی آن بخش به طور کامل کسر می‌شود.

7. تاریخ تحویل تمرین 1401/12/28، 11:59 شب، می‌باشد و سیاست‌های تاخیر مطابق با موارد ذکر شده در شیوه‌نامه لحاظ خواهد شد.

در صورت هرگونه ابهام در انجام تمرین با جناب آقای همتی از طریق ایمیل زیر در ارتباط باشید.

hrhemati@gmail.com