

انجام این تمرین به صورت انفرادی می باشد

## هدف از انجام این تمرین

هدف این تمرین مرور برخی از مفاهیم پایه مطرح شده در کلاس درس و همچنین آشنایی با مفاهیم پایه کنترل ربات است. برای انجام بخش پیاده سازی این تمرین، بهتر است در ابتدا ویدیو آماده شده برای این تمرین را مشاهده کنید.

## بخش های شبیه سازی:

### ❖ شرح سناریو

در این سناریو می خواهیم با کمک ROS و مباحث کنترلی که یاد گرفته ایم، سیستمی برای ربات خود طراحی کنیم که بتواند ربات را بر روی مسیر مورد نظر هدایت کند یا به یک مقصد دلخواه برساند. سپس به تحلیل سیستم طراحی شده و ارزیابی درستی عملکرد آن می پردازیم.

## ❖ گام اول :

در این گام، با کنترلر PID آشنا می شوید و بوسیله آن ربات را به سمت مقصد مورد نظر هدایت می کنید.

## ❖ بخش اول (۱۵ امتیاز) :

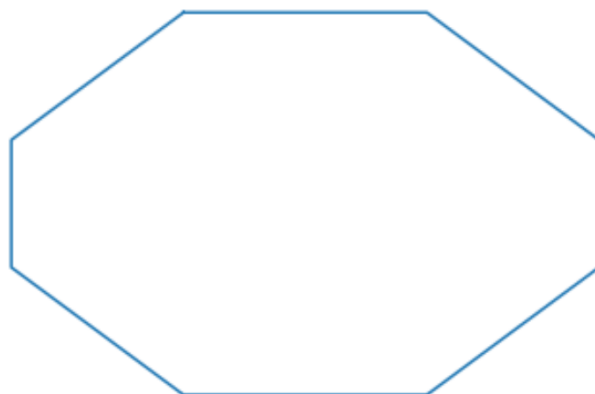
در این بخش می خواهیم تا ربات از نقطه شروع به سمت نقطه مقصد حرکت کند و در نهایت در مقصد متوقف شود. نقطه شروع ربات را مختصات (۰ و ۰) و مقصد را نیز مختصات (۴ و ۴) در نظر بگیرید. این کار را یک بار با کنترلر P یک بار با PD و بار دیگر با کنترلر PID انجام داده و نتایج را گزارش کنید.

## ❖ بخش دوم:

در این بخش، کنترلر باید به گونه ای طراحی شود که هر مسیر دلخواهی که در قالب یک آرایه ای از مختصات متوالی تعریف میشود را بتواند دنبال کند. برای این گام قصد داریم که ربات بتواند مسیر ها و شکل های متفاوتی را دنبال کند. می توانید برای تولید شکل های خواسته شده از کد قرار داده شده در [اینجا](#) استفاده کنید.

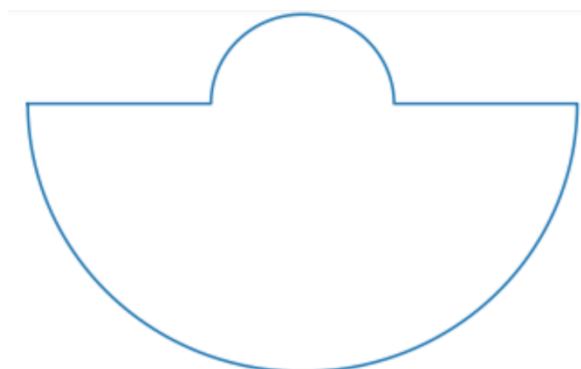
## • مسیر اول (۱۰ امتیاز) :

در این قسمت می خواهیم که ربات بر روی یک مسیر هشت ضلعی منتظم حرکت کند. ابتدا ربات باید به سمت نزدیکترین نقطه هشت ضلعی حرکت کند و حرکت خود را روی هشت ضلعی ادامه دهد، تا زمانی که برنامه متوقف شود. طول اضلاع هشت ضلعی را ۲ بگیرید. نقطه شروع ربات را یکبار بر روی مختصات (۰ و ۰) و بار دیگر بر روی مختصات (۱ و ۱) قرار دهید.



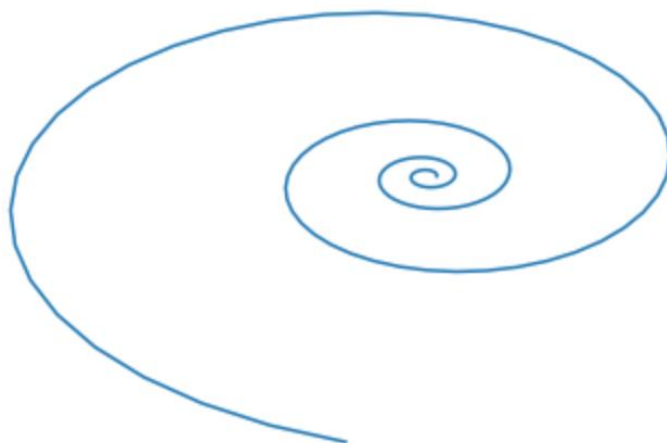
• مسیر دوم (۱۵ امتیاز):

در این قسمت ربات باید بر روی یک مسیر به شکل زیر حرکت کند، همانند قسمت قبل ربات حرکت خود را باید تا زمان توقف برنامه ادامه بدهد. نقطه شروع نیز، مختصات (۰ و ۰) می باشد.



• مسیر سوم (امتیازی):

در این قسمت ربات باید مسیری به شکل یک مارپیچ لگاریتمی با  $a = 17$  و مرکز (۰ و ۰) را دنبال کند. در ادامه می توانید تصویر این شکل را مشاهده کنید.



موارد زیر را در گزارش خود قرار دهید:

۱. نمای شکل تولید شده از حرکت ربات در شبیه ساز Rviz را نشان دهید

۲. تشریح کنید که چه ضرایبی برای  $P$ ،  $I$  و  $D$  پاسخ مناسبی را ارائه میکند. ( ضرایب مناسب را به صورت تجربی به دست آورید) و درباره تاثیرات افزایش و کاهش هر کدام از ضرایب بحث کنید و چند نمونه آن را تشریح کنید.

## ❖ گام دوم ( ۳۵ امتیاز ) :

در این مسئله ما دو ربات TurtleBot در صحنه خواهیم داشت. یکی از آن‌ها یعنی tb3\_0 توسط کیبورد و نود Teleop کنترل می شود و می توان به راحتی آن را در صحنه جابه جا کرد. برای ربات دیگر یعنی tb3\_1 یک کنترلر PID طراحی خواهیم کرد به طوری که این ربات دائماً در تعقیب ربات اول باشد و سعی کند تا فاصله‌ی خود را با آن کم کند.

برای این منظور پس از آنکه یک work space ساختید، در قسمت src می توانید به کمک دستورهای زیر پکیج‌های مورد نظر را دریافت نمایید:

- git clone -b noetic-devel [https://github.com/ROBOTIS-GIT/turtlebot3\\_simulations.git](https://github.com/ROBOTIS-GIT/turtlebot3_simulations.git)
- git clone -b noetic-devel <https://github.com/ROBOTIS-GIT/turtlebot3.git>
- git clone -b noetic-devel [https://github.com/ROBOTIS-GIT/turtlebot3\\_msgs.git](https://github.com/ROBOTIS-GIT/turtlebot3_msgs.git)

سپس فایل multi\_turtlebot3.launch که در مسیر زیر است را با فایل قرار داده شده در پوشه تمرین جایگزین نمایید:

- \turtlebot3\_simulations\turtlebot3\_gazebo\launch

همچنین فایل turtlebot3\_teleop\_key که در مسیر زیر است را با فایل قرار داده شده در پوشه تمرین جایگزین نمایید:

- \turtlebot3\turtlebot3\_teleop\nodes

پس از آن می‌توانید یک پکیج دیگر در کنار سایر پکیج‌ها در پوشه‌ی src مربوط به work space ایجاد کنید و کدهای کنترلر و لانچ فایل خودتان را در آن بنویسید. پس از این تغییرات و قبل از نوشتن کد مربوط به کنترلر می‌توانید به راحتی multi\_turtlebot3.launch را در یک ترمینال و teleop را در ترمینالی دیگر اجرا کنید و مشاهده نمایید که چگونه یکی از ربات‌ها حرکت خواهد کرد. در همین زمان می‌توانید در ترمینال دیگری تاپیک‌های موجود مثل tb3\_0/cmd\_vel/ را به کمک دستور rostopic list بررسی کنید و متوجه شوید که موقعیت ربات اول روی چه تاپیکی publish می‌شود. پس از انجام بررسی‌های لازم می‌توانید به راحتی نود کنترلر را برای ربات دوم طراحی کنید.

موارد زیر را در گزارش خود قرار دهید:

۱. ویدیوی کارکرد ربات

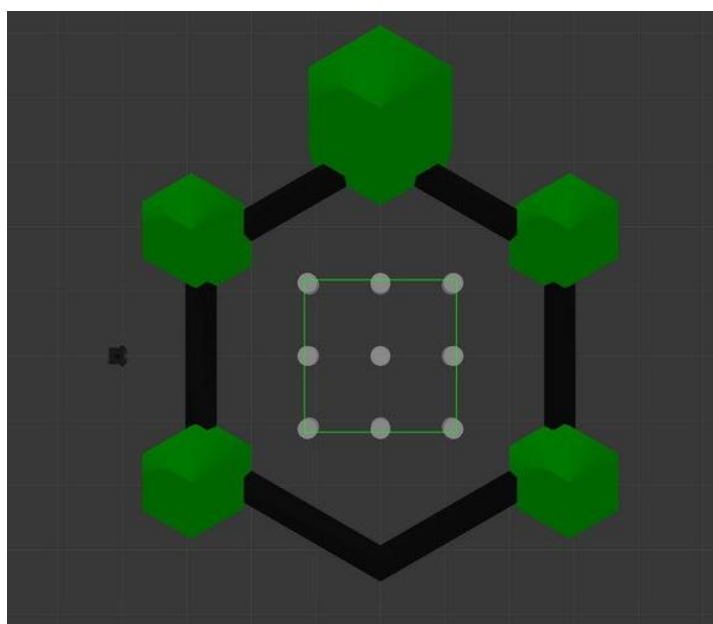
۲. نمای مسیر ربات دوم (tb3\_1) در شبیه ساز Rviz

## ❖ گام سوم (۲۵ امتیاز) :

در این سوال باید یک الگوریتم دنبال کردن دیوار (following wall) را با کمک سنسور LaserScan پیاده سازی کنید. این الگوریتم ساده برای حرکت ابتدایی در یک محیط ناشناخته مناسب می باشد.

بر اساس این الگوریتم، ربات همواره در کنار دیوار حرکت میکند. ربات می تواند دنبالگر دیوار راست یا چپ باشد. دنبالگر دیوار راست، در هر لحظه دیوار را در سمت راست خود میبیند و حرکت میکند. به این صورت که هنگامی که ربات مستقیم حرکت کرده و دیوار را پیدا نمود، به سمت چپ می پیچد تا دیوار در سمت راست آن قرار بگیرد. اگر در ادامه مسیر ربات در مقابلش به دیوار برسد، و همچنان در راست خود دیوار باشد، به چپ می پیچد. اما هنگامی که دیوار سمت راست پایان می یابد، به راست گردش می کند تا دوباره دیوار در سمت راست خود قرار گیرد و سپس حرکت میکند. ربات دنبالگر چپ نیز برخلاف این دنبالگر همواره دیوار سمت چپ خودش را دنبال میکند.

در این قسمت شما می توانید دنبالگر راست، و یا چپ را پیاده سازی نمایید. ربات همواره یک فاصله مشخصی را با دیوار حفظ کرده و آن را دنبال میکند. این فاصله را خودتان تعیین کنید. ربات می تواند با سرعت خطی حرکت داشته باشد و زمانی که نیاز به چرخش باشد، بایستد، و بچرخد. (البته لزومی به ایستادن ندارد و می تواند سرعت خطی ثابتی داشته باشد ولی دقت شود که خطا هنگام پیچیدن زیاد نباشد.) دقت کنید چرخش ربات لزوماً ۹۰ درجه نمی باشد و وابسته به فاصله ای که ربات با دیوار دارد میباشد. یعنی چرخش تا زمانی که با دیوار فاصله مشخص داشته باشد، ادامه می یابد. ابتدا ربات را در دنیای turtlebot3\_world.world به سمت دیوار قرار دهید. ربات باید یک مسیر مستقیم را طی کند تا به یک دیوار برسد. سپس دیوار را دنبال نماید و کل مسیر را طی کند. همچنین می توانید نقطه شروع ربات را مختصات (0, 4) و با زاویه اولیه 1.57- در نظر بگیرید.



موارد زیر را در گزارش خود قرار دهید:

۱. نمای شکل تولید شده از حرکت ربات در شبیه ساز Rviz را نشان دهید
۲. تشریح کنید که چه ضرایبی برای  $P$ ،  $I$  و  $D$  پاسخ مناسبی را ارائه می‌کند.

#### ❖ نکات :

- ✓ برای تمامی گام ها، اجرا را حداقل به گونه ای در نظر بگیرید که در داخل تصویر گرفته شده از مسیر طی شده شکل خواسته شده کاملاً قابل تشخیص باشد.
- ✓ برای انجام این تمرین، کنترلی که برای سرعت خطی طراحی میکنید باید PID باشد، هرچند برای سرعت زاویهای هر کنترلی را میتوانید طراحی کنید.

#### ❖ 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.

### نکات تکمیلی در باب تحویل تمرین

۱. تحویل گزارش در یک فایل pdf مطابق با قالب قرار گرفته در سایت کورسز و با نام‌گذاری HW3\_StudentNumber می‌بایست تحویل داده شود.
۲. فرمت فایل زیپ نیز به صورت HW3\_StudentNumber باشد.
۳. افراد می‌بایست تمرین را به صورت انفرادی انجام دهند.
۴. 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.
۵. برای انجام تمرین و پروژه تنها زبان برنامه‌نویسی مجاز پایتون می‌باشد.
۶. 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.
۷. تاریخ تحویل تمرین ۱۴۰۳/۲/۲۱، ۱۱:۵۹ شب، می‌باشد و سیاست‌های تاخیر مطابق با موارد ذکر شده در شیوه‌نامه لحاظ خواهد شد.