Task 1A

Note: Deadline of Task 1a and Task 1b is 10th October 2023

Task 1A is a relatively simple task designed just for you to get comfortable with the usage of ROS2

*Let's directly jump onto the **Problem Statement** without wasting any time (we have a deadline to catch!).*

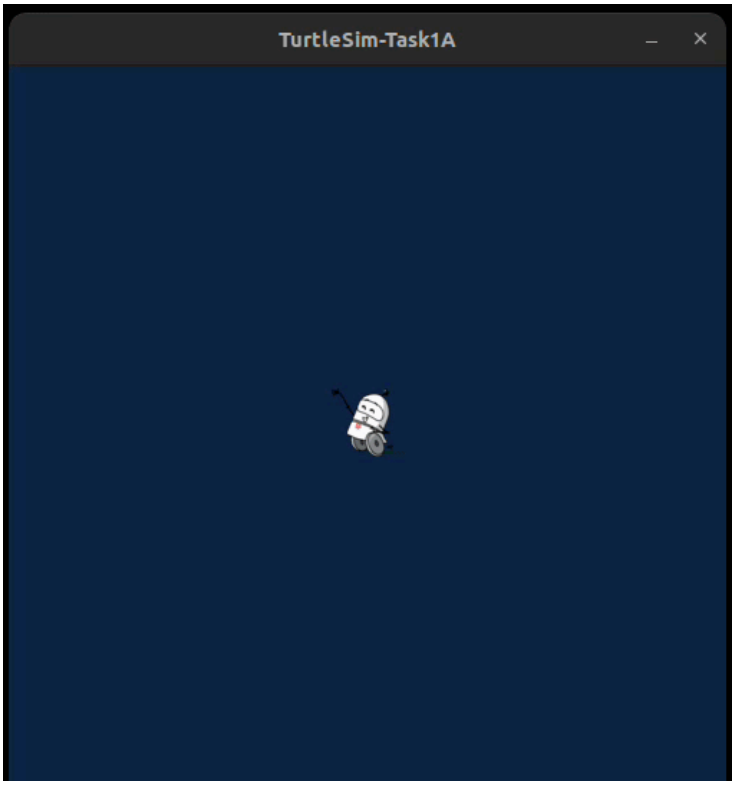
Problem Statement

Create a simple controller for **turtlesim** using python and use it to perform the desired maneuver with 2 turtles (or e-YAN???) exactly as described below.

There are three things that need to be done:

- First, draw a circle with the first turtle.
- When the first circle is drawn, the controller node must stop the first turtle and call a service to spawn another turtle.
- Lastly, the second turtle should draw another circle bigger than the previous one and stop when it is drawn.

The final output/drawing should resemble the appearance of a snowman. ⛄
The expected output for Task 1A is given below 👇 :



Expected Output

Approach

So how shall we go about implementing task 1A?

Let's complete the following steps one by one:

- **Step 1: Creating a ROS2 Workspace**
  Since we will be working with custom packages and customised nodes, we will be creating a workspace where all our changes get incorporated at once when we source the specific workspace. We recommend to create a directory named `eyrc_hb` where all your different ROS2 workspaces will reside. (Yes, we will have a different workspaces for different tasks)

Now open the terminal and run the following commands step by step:

```
sudo apt update
```

```
cd
```

Create a new folder **eyrc_hb** and create another folder named **hb_task1a_ws** which will be our workspace.

(*in `-p` mode, `mkdir` command creates the parent directories if not already created*)

```
mkdir -p eyrc_hb/hb_task1a_ws/src
```

Navigate to the workspace folder

```
cd eyrc_hb/hb_task1a_ws
```

Build a workspace. After this command, your **hb_task1a_ws** will become a ROS2 workspace.

```
colcon build --symlink-install
```

- **Step 2: Creating a new package inside the ROS2 Workspace**

Now navigate to the `src` folder using `cd src` inside the workspace and run the following command:

```
ros2 pkg create --build-type ament_python hb_task_1a
```

Now you have successfully created a new package. As you learnt in the ros2 learning resources, in this package we will create the **task_1a** node that will be called to perform the desired task!

To create the node, navigate to the `hb_task_1a` folder inside `/src/hb_task_1a` and run the following commands:

```
touch task_1a_<team_id>.py
```

```
chmod +x task_1a_<team_id>.py
```

where <team_id> is the is your team ID. For example, if your team ID is 9999, you should create the file names `task_1a_9999.py` .
Do not usse 9999 as your team ID, this is just an example

The `touch` command creates a python file and the `chmod` command converts it into an executable.

Before you continue to build the package further, download the following .zip file and extract it in the src folder of your workspace.

TurtleSim Customized package (*ros_tutorials.zip*)

- **Step 3: Editing the task_1a_9999.py file**

Open the `task_1a_9999.py` file in your favourite code editor. (We highly recommend VS Code!)

Now edit the `task_1a_9999.py` file and implement your logic. We have provided an example in learning resources to get you thoroughly acquainted.

Refer to ROS2 Publisher-Subscriber and Service Example (Python)

Do not forget to add the dependencies in the `package.xml` file and add the new nodes in the `setup.py` file inside the `'console scripts' : []` list inside the `entry_points` dictionary!

You can refer this document for reference- ROS2 Wiki: Creating a Package

And remember to always **save** your changes!!

Note: Please paste the following comments in your `task_1a_9999.py` file and enter the required info. as per your team. Also, make sure the coding standards are adhered.

```
##############################################################################
########################################## eYRC 23-24 Hologlyph Bots Task 1A ##########
# Team ID:
# Team Leader Name:
# Team Members Name:
# College:
##############################################################################
```
