

AA 274A: Principles of Robot Autonomy I

Section 1

Name: Pol Francesch Huc
SUID: 006650276

Problem 1

Determine the following things about the server.

- a **How many GPU's are there?**
There are 6 GPU's.
- b **How much RAM is available on the machine?**
There are 252 GB of RAM.
- c **How many CPU cores are there?**
There are 64 cores.
- d **What version of Python is available on the machine?**
Python version 3.8.10 is available as alias python3.

Problem 2

Create your own message file consisting of multiple standard data types.

```
# Message used to pass information about a specific latitude, longitude.
string  landType
float64 latitude
float64 longitude
float64 altitude
```

Problem 3

Create a publisher and subscriber that publish and subscribe to your custom message type, respectively.

publisher.py

```
import rospy
from aa274a_s2.msg import MyMessage

def publisher():
    pub = rospy.Publisher('my_topic', MyMessage, queue_size=10)
    rospy.init_node('my_node', anonymous=True)
    rate = rospy.Rate(1)
    msg = MyMessage()
    while not rospy.is_shutdown():
        msg.landType = "Building"
        msg.latitude = 37.429994
        msg.longitude = -122.173248
        msg.altitude = 25
        pub.publish(msg)
        rate.sleep()

if __name__ == '__main__':
    try:
        publisher()
    except rospy.ROSInterruptException:
        pass
```

subscriber.py

```
import rospy
from aa274a_s2.msg import MyMessage

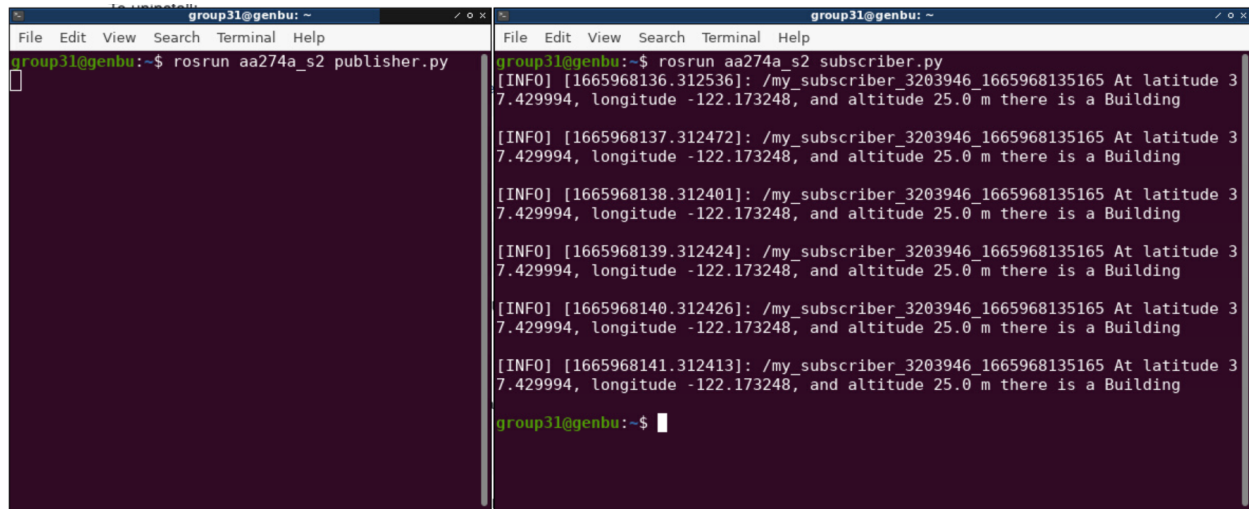
def callback(data):
    msg = " At latitude " + str(data.latitude)
    msg = msg + ", longitude " + str(data.longitude)
    msg = msg + ", and altitude " + str(data.altitude)
    msg = msg + " m there is a " + str(data.landType)
    rospy.loginfo(rospy.get_caller_id() + msg + "\n")

def subscriber():
    rospy.init_node('my_subscriber', anonymous=True)
    rospy.Subscriber("my_topic", MyMessage, callback)
    rospy.spin()

if __name__ == '__main__':
    subscriber()
```

Problem 4

Include screenshots or terminal output text that shows your publisher and subscriber are working.



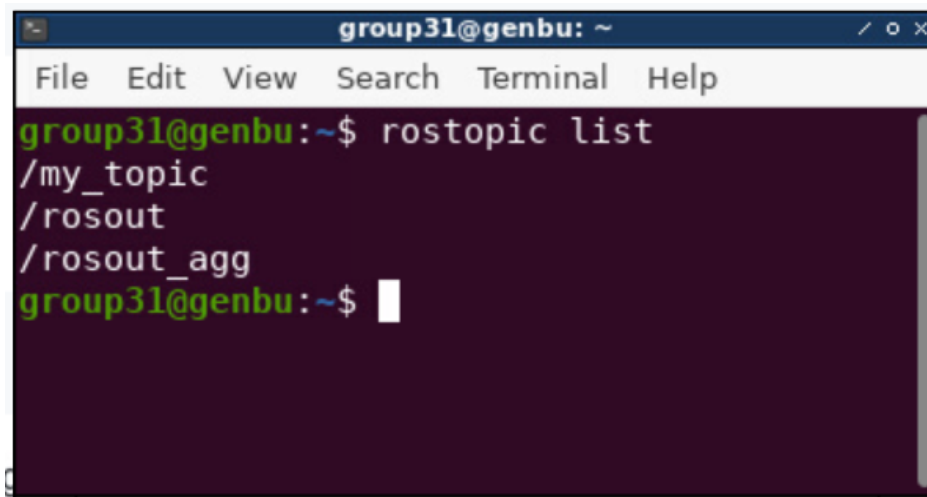
The image shows two terminal windows side-by-side. The left window has the title 'group31@genbu: ~' and shows the command 'rosrun aa274a_s2 publisher.py' being executed. The right window also has the title 'group31@genbu: ~' and shows the command 'rosrun aa274a_s2 subscriber.py' being executed. It displays six lines of INFO messages from a subscriber, each indicating a building at a specific latitude and longitude.

```
group31@genbu: ~  
File Edit View Search Terminal Help  
group31@genbu:~$ rosrun aa274a_s2 publisher.py  
  
group31@genbu: ~  
File Edit View Search Terminal Help  
group31@genbu:~$ rosrun aa274a_s2 subscriber.py  
[INFO] [1665968136.312536]: /my_subscriber_3203946_1665968135165 At latitude 3  
7.429994, longitude -122.173248, and altitude 25.0 m there is a Building  
  
[INFO] [1665968137.312472]: /my_subscriber_3203946_1665968135165 At latitude 3  
7.429994, longitude -122.173248, and altitude 25.0 m there is a Building  
  
[INFO] [1665968138.312401]: /my_subscriber_3203946_1665968135165 At latitude 3  
7.429994, longitude -122.173248, and altitude 25.0 m there is a Building  
  
[INFO] [1665968139.312424]: /my_subscriber_3203946_1665968135165 At latitude 3  
7.429994, longitude -122.173248, and altitude 25.0 m there is a Building  
  
[INFO] [1665968140.312426]: /my_subscriber_3203946_1665968135165 At latitude 3  
7.429994, longitude -122.173248, and altitude 25.0 m there is a Building  
  
[INFO] [1665968141.312413]: /my_subscriber_3203946_1665968135165 At latitude 3  
7.429994, longitude -122.173248, and altitude 25.0 m there is a Building  
group31@genbu:~$
```

Problem 5

Run the first three of these commands to:

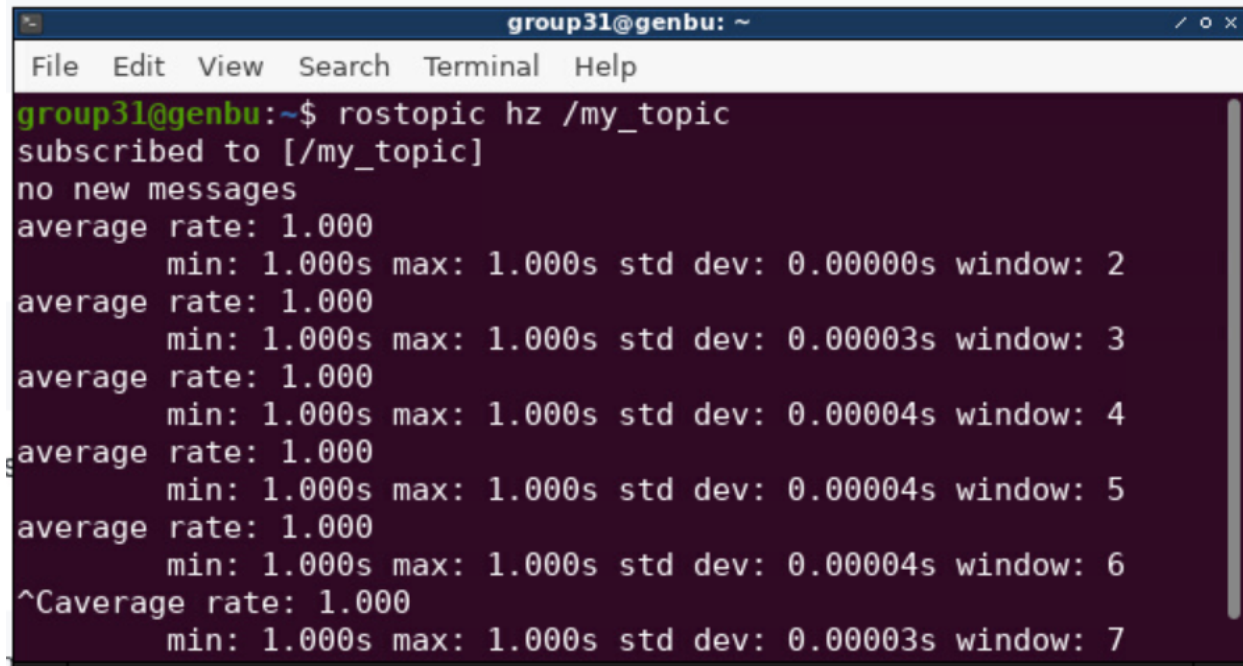
- See that your topic is registered and visible.



The image shows a terminal window with the title 'group31@genbu: ~'. It displays the command 'rostopic list' and its output, which lists three topics: '/my_topic', '/rosout', and '/rosout_agg'.

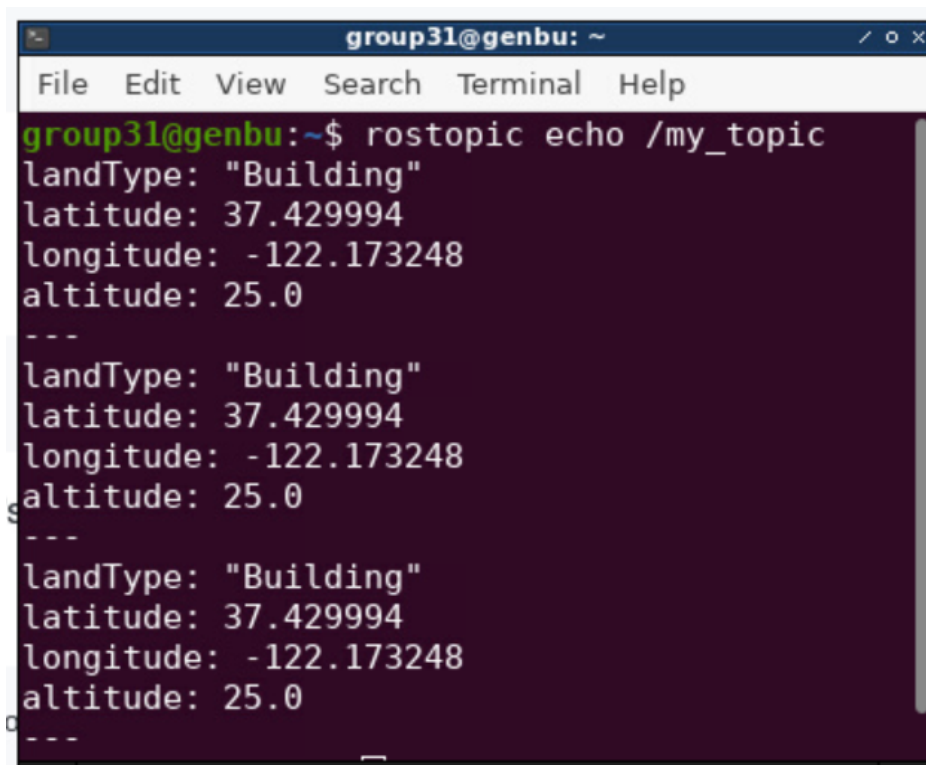
```
group31@genbu: ~  
File Edit View Search Terminal Help  
group31@genbu:~$ rostopic list  
/my_topic  
/rosout  
/rosout_agg  
group31@genbu:~$
```

- b Show what your publisher is publishing.

A terminal window titled 'group31@genbu: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The command 'rostopic hz /my_topic' has been executed. The output shows the subscription status and a series of frequency statistics for the /my_topic topic. The statistics are repeated for window sizes 2 through 7, all showing an average rate of 1.000 Hz and a minimum time of 1.000s.

```
group31@genbu:~$ rostopic hz /my_topic
subscribed to [/my_topic]
no new messages
average rate: 1.000
  min: 1.000s max: 1.000s std dev: 0.00000s window: 2
average rate: 1.000
  min: 1.000s max: 1.000s std dev: 0.00003s window: 3
average rate: 1.000
  min: 1.000s max: 1.000s std dev: 0.00004s window: 4
average rate: 1.000
  min: 1.000s max: 1.000s std dev: 0.00004s window: 5
average rate: 1.000
  min: 1.000s max: 1.000s std dev: 0.00004s window: 6
^Caverage rate: 1.000
  min: 1.000s max: 1.000s std dev: 0.00003s window: 7
```

- c Determine the frequency with which your publisher is publishing messages.

A terminal window titled 'group31@genbu: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The command 'rostopic echo /my_topic' has been executed. The output shows three identical JSON-like messages being received on the /my_topic topic. Each message contains fields for landType, latitude, longitude, and altitude.

```
group31@genbu:~$ rostopic echo /my_topic
landType: "Building"
latitude: 37.429994
longitude: -122.173248
altitude: 25.0
---
landType: "Building"
latitude: 37.429994
longitude: -122.173248
altitude: 25.0
---
landType: "Building"
latitude: 37.429994
longitude: -122.173248
altitude: 25.0
---
```