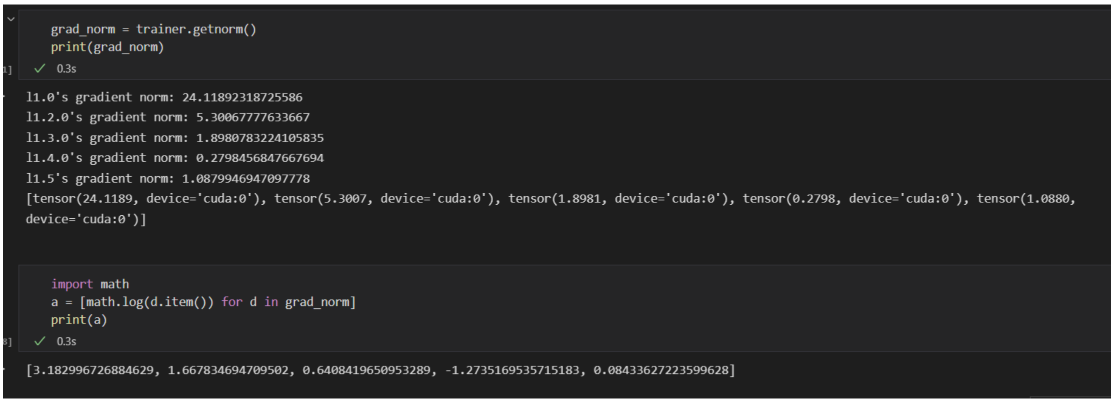
1. Describe the difference between WGAN* and GAN**, list at least two differences
   (1) In WGAN, discriminator does not have sigmoid function.
   (2) In WGAN, it uses a new loss function derived from the Wasserstein distance, and we do not take a logarithm of the loss function in both generator and discriminator
   (3) In WGAN, it clamp the weights to a small fixed range [-c,c] after every gradient update on the critic function.
   (4) In WGAN, the author recommend RMSProp optimizer rather than Adam.

2. Please plot the “Gradient norm” result.



```
grad_norm = trainer.getnorm()
print(grad_norm)
```

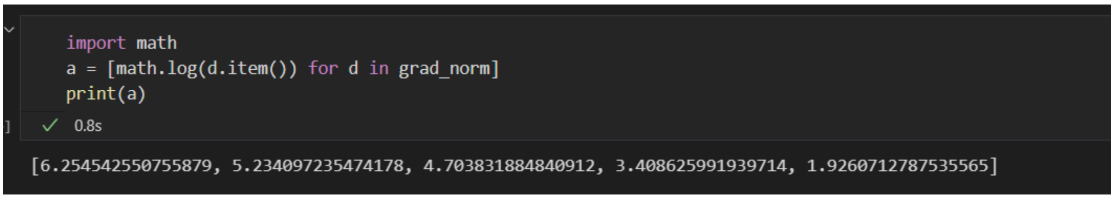
✓ 0.3s

```
l1.0's gradient norm: 24.11892318725586
l1.2.0's gradient norm: 5.30067777633667
l1.3.0's gradient norm: 1.8980783224105835
l1.4.0's gradient norm: 0.2798456847667694
l1.5's gradient norm: 1.0879946947097778
[tensor(24.1189, device='cuda:0'), tensor(5.3007, device='cuda:0'), tensor(1.8981, device='cuda:0'), tensor(0.2798, device='cuda:0'), tensor(1.0880,
device='cuda:0')]
```

```
import math
a = [math.log(d.item()) for d in grad_norm]
print(a)
```

✓ 0.3s

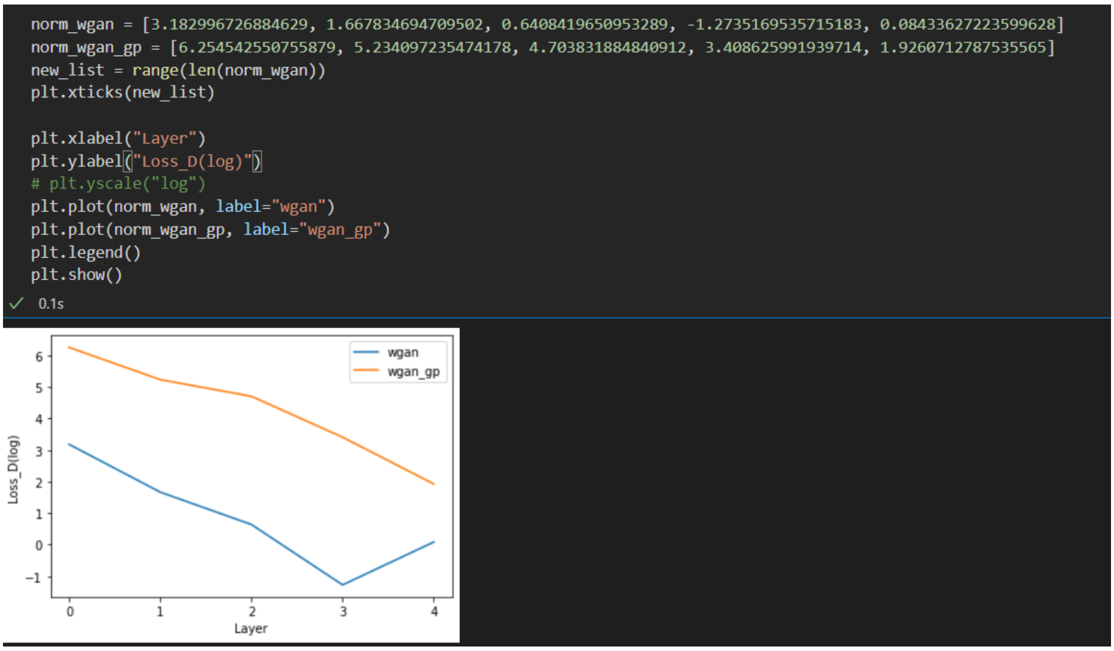
```
[3.182996726884629, 1.667834694709502, 0.6408419650953289, -1.2735169535715183, 0.08433627223599628]
```



```
import math
a = [math.log(d.item()) for d in grad_norm]
print(a)
```

✓ 0.8s

```
[6.254542550755879, 5.234097235474178, 4.703831884840912, 3.408625991939714, 1.9260712787535565]
```



```
norm_wgan = [3.182996726884629, 1.667834694709502, 0.6408419650953289, -1.2735169535715183, 0.08433627223599628]
norm_wgan_gp = [6.254542550755879, 5.234097235474178, 4.703831884840912, 3.408625991939714, 1.9260712787535565]
new_list = range(len(norm_wgan))
plt.xticks(new_list)

plt.xlabel("Layer")
plt.ylabel("Loss_D(log)")
# plt.yscale("log")
plt.plot(norm_wgan, label="wgan")
plt.plot(norm_wgan_gp, label="wgan_gp")
plt.legend()
plt.show()
```

✓ 0.1s

ps: Y 軸 label 應為 gradient norm, 誤植為 loss