

HOMWORK 5 TEMPLATE

Use this template to record your answers for Homework 5. Add your answers using L^AT_EX and then save your document as a PDF to upload to Gradescope. You are required to use this template to submit your answers. **You should not alter this template in any way** other than to insert your solutions. You must submit all ?? pages of this template to Gradescope. Do not remove the instructions page(s). Altering this template or including your solutions outside of the provided boxes can result in your assignment being graded incorrectly.

You should also export your code as a .py file and upload it to the **separate** Gradescope coding assignment. Remember to mark all teammates on **both** assignment uploads through Gradescope.

Instructions for Specific Problem Types

On this homework, you must fill in blanks for each problem. Please make sure your final answer is fully included in the given space. **Do not change the size of the box provided.** For short answer questions you should **not** include your work in your solution. Only provide an explanation or proof if specifically asked.

Fill in the blank: What is the course number?

10703

Problem 0: Collaborators

Enter your team members' names and Andrew IDs in the boxes below. If you worked in a team with fewer than three people, leave the extra boxes blank.

Name 1:	Andrew Jong	Andrew ID 1:	ajong
Name 2:	Gabe Olin	Andrew ID 2:	golin
Name 3:	Mike Anoruo	Andrew ID 3:	manoruo

Problem 1: Model-Based Reinforcement Learning with PETS (105 pt)

1.1 Model-based Predictive Control (25 pts)

1.1.1 CEM (without MPC) with ground-truth dynamics (10 pt)

Success percentage

80%

1.1.2 Random sampling with ground-truth dynamics. (10 pt)

Success percentage without MPC

68%

Success percentage with MPC

84%

How does the performance of random sampling performance compare to that of CEM?

CEM performs better than random sampling in my experiments. CEM without MPC achieved an 80% success rate, while random sampling without MPC only reached 68%. With MPC, random sampling improves to 84%, but CEM still outperforms it. This shows CEM's iterative refinement helps find better action sequences, leading to higher success rates compared to random sampling.

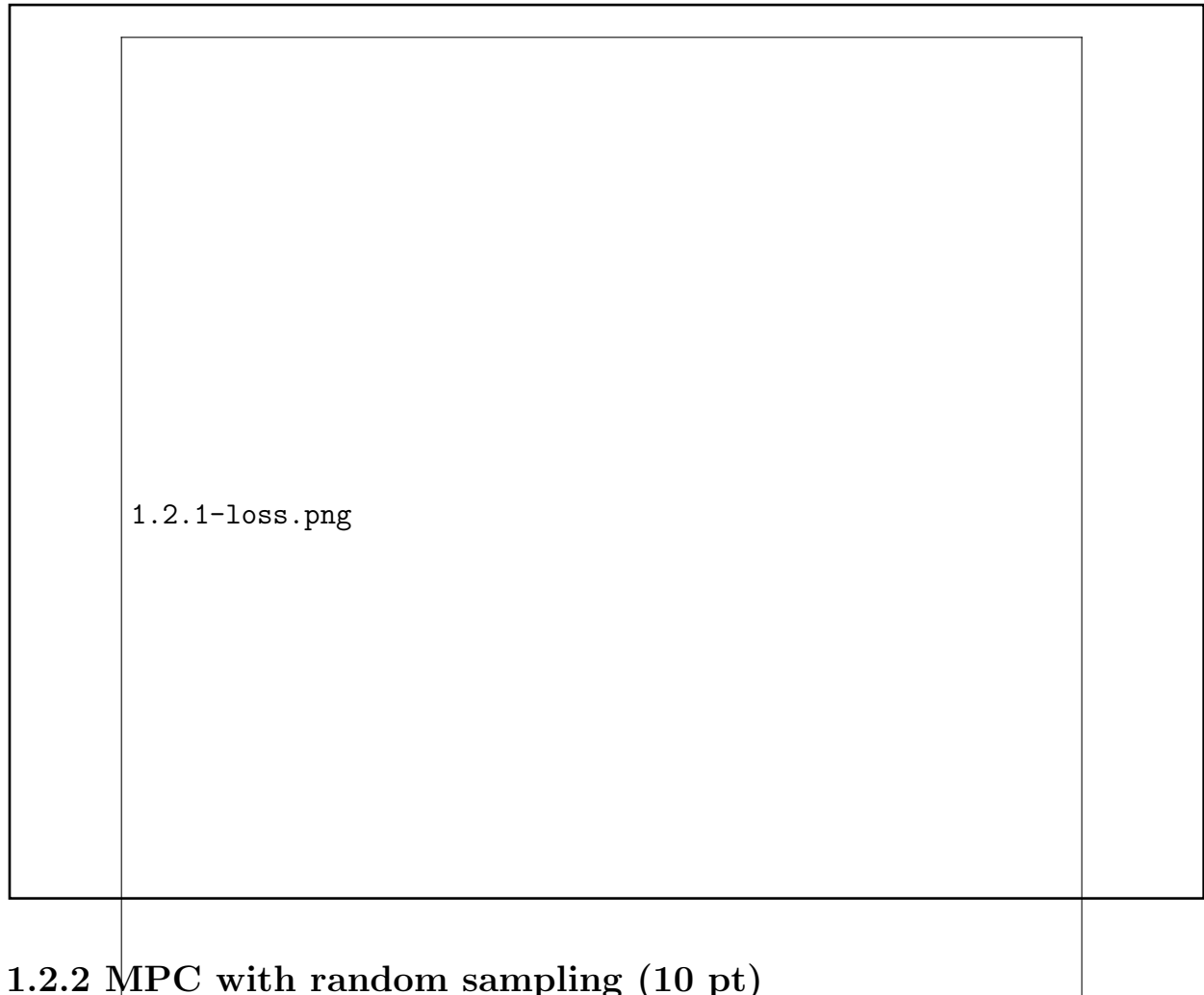
1.1.3 MPC vs. open-loop control (5 pt)

<p>MPC performs better than open-loop control in dynamic/changing environments because it uses feedback from real time observations to correct its actions at each step. MPC adapts to disturbances and model inaccuracies by replanning continuously, making it a more robust algorithm. Open-loop control works better in stable/predictable environments where a precomputed action sequence is sufficient.</p>
--

<p>The pros of MPC include its ability to handle disturbances while optimizing future behavior; however, a con is that it comes with higher computational costs and relies on accurate modeling.</p>
--

1.2 Single probabilistic network (30 pts)

1.2.1 Training loss plot (4 pt)



1.2.2 MPC with random sampling (10 pt)

Success percentage

1.2.3 MPC with CEM (10 pt)

Success percentage

1.2.4 Random sampling vs. CEM (6 pt)

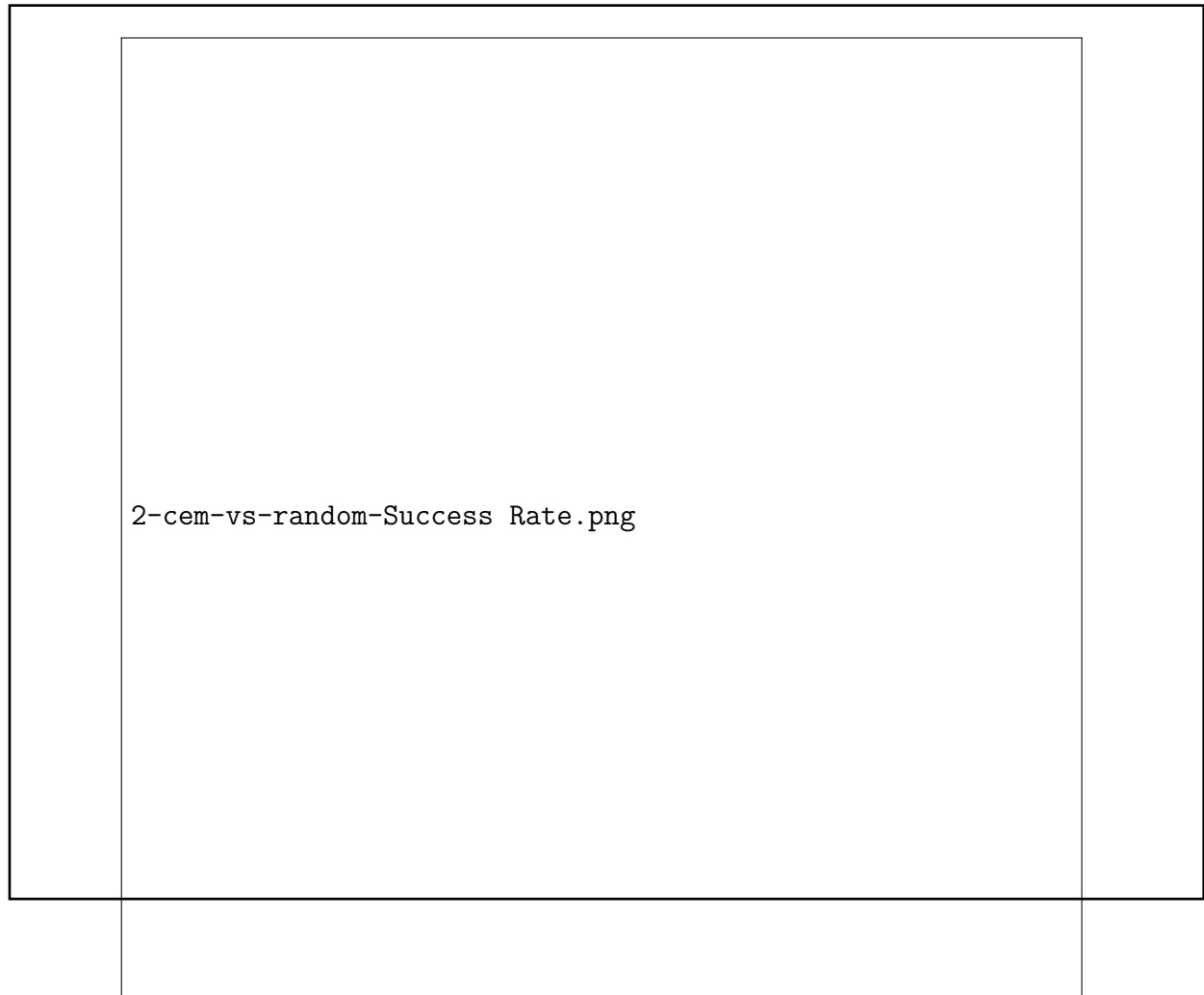
CEM outperforms random action by more intelligently selecting action sequences that lead to higher success rates. CEM + MPC didn't perform great in the recorded performance, achieving a success rate of only 36%. However, on varying runs, it shows promise. The derived policy fails when it sees out-of-distribution state-action pairs. Our single probabilistic model starts to predict inaccurate transitions, and this error amplifies over longer horizons.

1.3 MBRL with PETS (15 pts)

1.3.1 Training loss plot (10 pt)

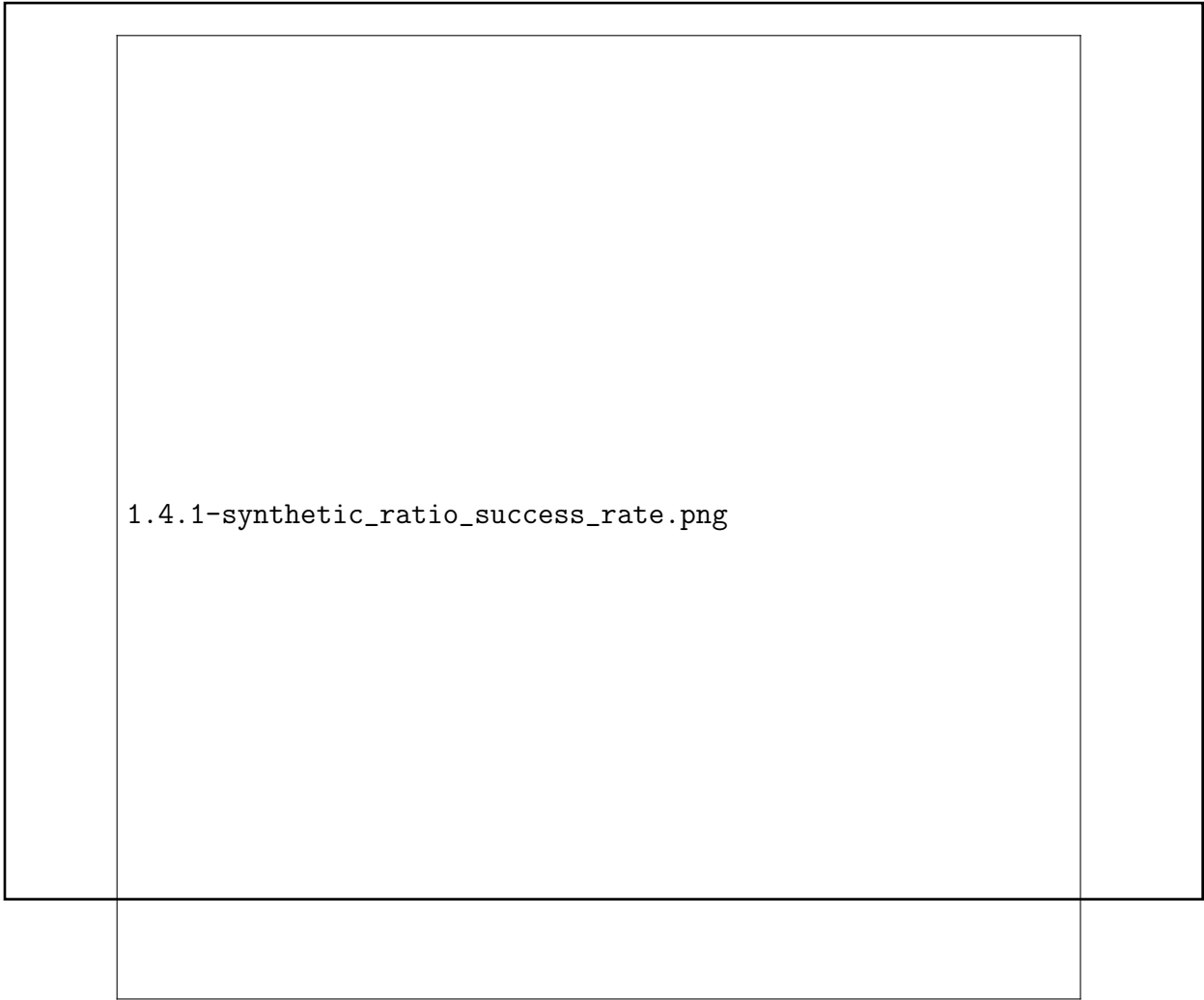
1.3.1-loss.png

1.3.2 Test percentage of successes plot (5 pt)



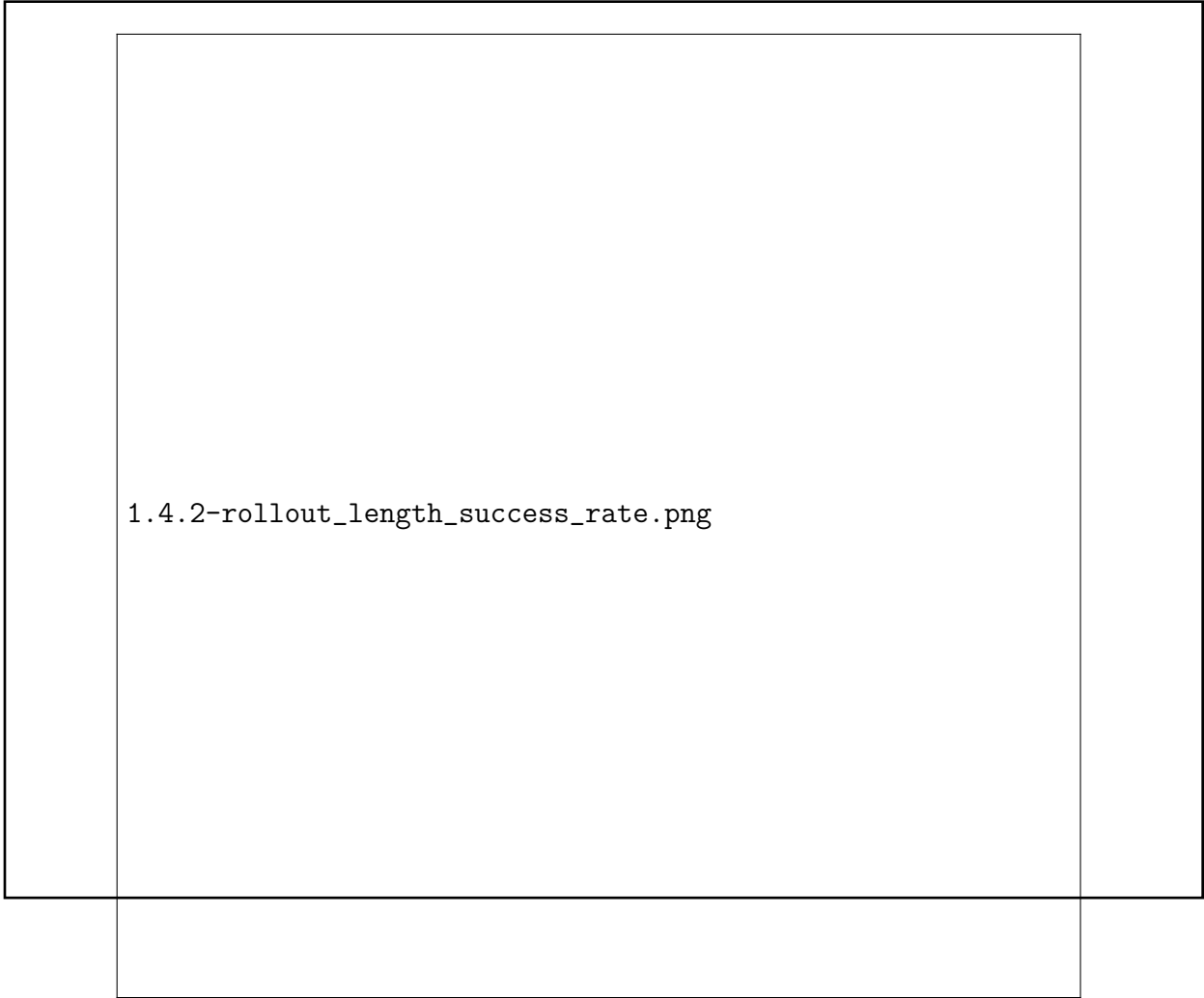
1.4 Training with synthetic data (30 pts)

1.4.1 Ablation over model vs real data (10 pt)



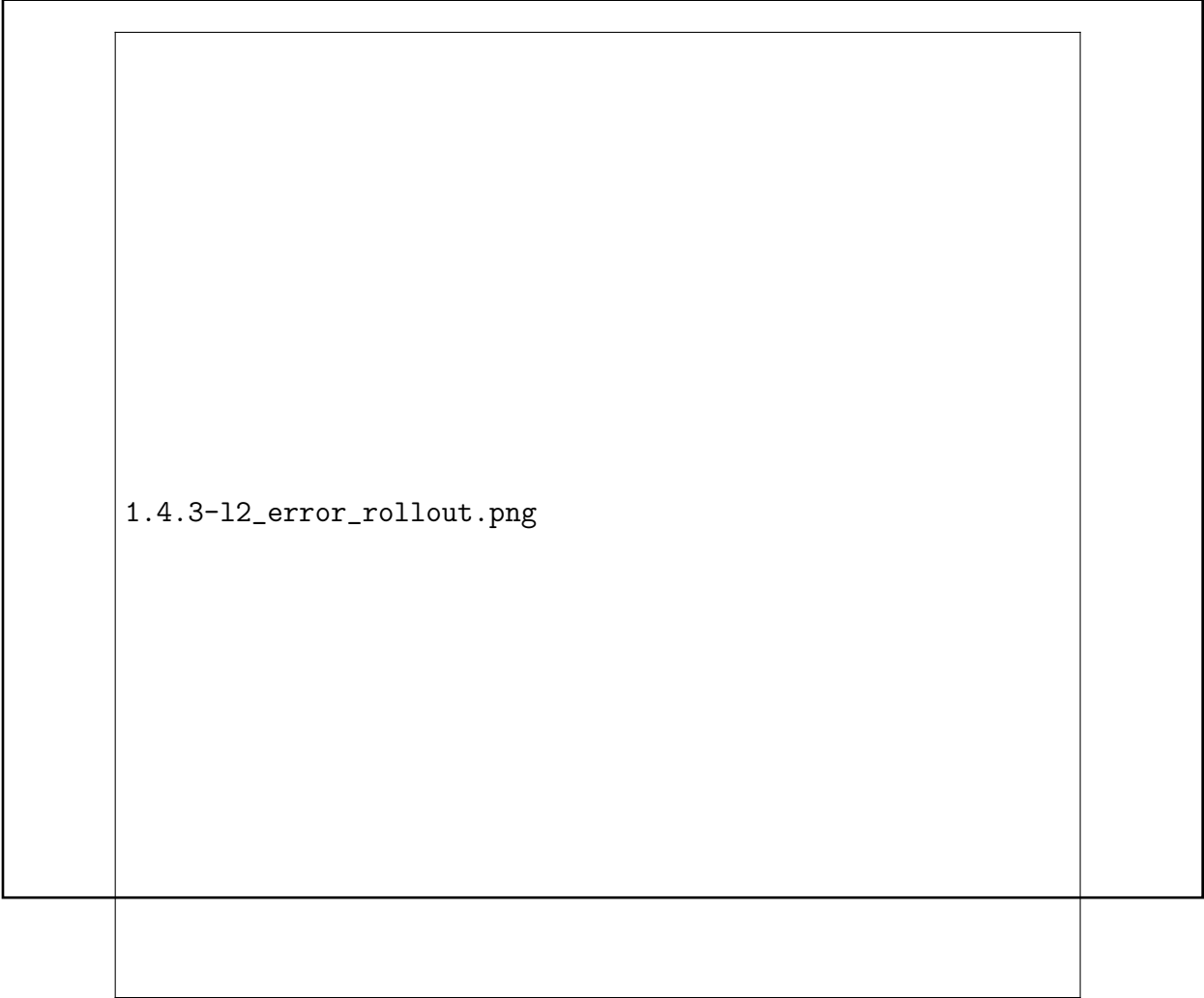
1.4.1-synthetic_ratio_success_rate.png

1.4.2 Ablation over the rollout lengths (10 pt)



1.4.2-rollout_length_success_rate.png

1.4.3 Model error over rollout lengths (10 pt)



1.4.3-12_error_rollout.png

1.4.4 MBRL vs Policy Gradient (5pt)

MBRL is more preferable when sample efficiency and fast adaptation are important, as it uses a learned model to simulate and plan without excessive "real world" interaction. It also excels in tasks requiring planning and foresight. Policy gradients are preferred when modeling the environment is difficult or infeasible, especially in high-dimensional or complex action spaces.

Feedback

Feedback: You can help the course staff improve the course for future semesters by providing feedback. You will receive a point if you provide actionable feedback. What was the most confusing part of this homework, and what would have made it less confusing?

Solution

The documentation could be made a lot easier to follow with a clearer structure and consistent numbering. It sometimes felt confusing to jump between parts because it wasn't always obvious how the sections connected. A bit more organization and a smoother flow could help.

Collaboration: Detail the work division amongst your group below.

Solution

All members contributed equally. We worked on different parts of the code separately, but regularly discussed and helped each other fix bugs.

Time Spent: How many hours did you spend working on this assignment? Your answer will not affect your grade. Please average your answer over all the members of your team.

Solution

Alone	6
With teammates	2
With other classmates	0
At office hours	2