This document provides guidance on the types of questions and analyses we expect to find in your Mini-project reports and presentations. These are just examples, so please feel free to add analysis and discussion based on what is relevant and interesting about your specific approach.

Overall goal

The main purpose of this Mini-project is to explore the challenges that we and all animals face in performing complex hierarchical control tasks. It inspires us to consider how to implement controllers to solve similar problems in the autonomous control of robots.

Global questions (to be addressed for each Task Level)

- **Describe the control strategies** that you used to try to solve this Level.
- Explain which strategies **did not work** and why they likely did not work (e.g., failure modes encountered in certain seeds).
- Explain which strategies **did work** and why you think they worked.
- Quantify to what extent specific parts of your strategy contributed to task success by comparing performance **with or without** including this part of the control strategy.

Specific questions (to be addressed for this particular Task Level)

Locomotion and turning

- What locomotor controller did you use to navigate over the terrain?
- What descending signal did you use to control turning?
    - Why did you use this approach?
- How did your control approach differ (if at all) from the ones we implemented in Exercise Week 2 and 3?
    - Why did you implement these changes how did they improve performance?

- What are some ideas you have to improve this performance (but did not have the time to implement)?

Level 0. Empty Arena - An odor source must be reached

- What odor taxis strategy did you implement?
    - How did it differ (if at all) from the one we implemented in Exercise Week 3?
    - Why did you implement these changes how much (quantify) did they improve performance?

- What are some ideas you have to improve this performance (but did not have the time to implement)?

Level 1. Pillars Arena

- What sensory input(s) did you use to identify the pillars? Visual? Tactile? Why did you select this strategy?

- If visual, which NeuroMechFly visual inputs did you use to detect the pillars? Raw camera images or retina/ommatidia readings?
    - What features (distance, direction, absence of pillars) did you extract from the visual inputs? How did you extract them?
    - How robust is this approach to noisy visual sensor readings?

- If tactile, which specific NeuroMechFly sensors did you use to detect touch? Why did you select these sensors specifically?
  - What approach did you come up with to process sensor signals to detect obstacles using mechanosensation?
  - How effective is this approach for detecting the left/right position of an obstacle?
  - How robust is this approach to noisy mechanosensor readings?

- Once pillars are detected by your algorithm how are they then avoided during navigation? For example, does your controller need to drive backward turning or only turning?
  - What other information did you use (besides obstacle detection) in your algorithm to control navigation around obstacles? E.g., did you take into account the leg's state (swing/stance)?

- What are some ideas you have to improve this performance (but did not have the time to implement)?

Level 2. Looming Balls Arena

- Which NeuroMechFly visual inputs did you use to detect the looming balls? Raw camera images or retina/ommatidia readings?
  - How did you detect visual looming (versus other visual stimuli like pillar movements due to self-motion)?
  - How did you estimate the direction from which the looming was approaching from ommatidia readings?
  - How did you estimate the speed of looming to anticipate how long it would take for a ball to arrive?

- What behavioral strategies did you use to avoid balls approaching from different directions? Reversal? Turning away? Speeding forward? Something else?
  - Why did you select this approach?

- What visual features of the looming ball were used to trigger the reaction towards it (e.g., size in visual field, velocity through visual field…)?
  - How did you use these features to adjust the descending signal of the fly?
  - How robust is this approach to noisy visual sensor readings?

- What are some ideas you have to improve this performance (but did not have the time to implement)?

Level 3. Pillars and Looming Balls Arena

- If you used vision for both, how did you parse out the signals from the pillars versus signals from the looming balls?
  - With two visual tasks to solve simultaneously, how does your controller choose to prioritize one versus the other?
  - How robust was this approach in different starting seeds of the task?

- What are some ideas you have to improve this performance (but did not have the time to implement)?

Level 4. Path integration arena

- What path integration strategy did you implement?
  - How did it differ (if at all) from the one we implemented in Exercise Week 6?
  - Why did you implement these changes how much (quantify) did they improve performance?

- What are some ideas you have to improve this performance (but did not have the time to implement)?

Bonus. Biological realism

- Did you try to implement components of your controller that were more biologically inspired (e.g., connectome vision model)?
  - If so, what did you learn about biological approaches compared to traditional approaches?

- If you used the connectome-based visual system, which connectome-based network neurons were most informative for the visual features you cared about to detect pillars or looming balls?
  - How did you combine this array of visual neurons to discriminate between pillars and looming balls?
  - How did you combine this array of visual neurons to drive the descending signal?

- What are some ideas you have to improve this performance (but did not have the time to implement)?