

# CLASS SET

## "Oh, Deer" Evaluation Questions

1. Make a graph of the data from PART 1. Be sure to label the "x" and "y" axes. Title the graph.
2. Make a different graph of the data from PART II. Graph both the deer and wolves on the same graph. Be sure to label the "x" and "y" axes, title the graph, and provide a key.

### Answer the following:

#### Part I: Deer and Resources

3. Which trophic level do the deer represent?
4. Why did the resources become deer after being collected by a deer? (What did this represent?)
5. When the deer population increased to a high level, what happened in the next round? Why?
6. Which parts of the game represented the limiting factors? Were these factors density-dependent or density-independent? Explain.
7. When deer populations decreased greatly, what happened in the next round? Why?
8. Make a graph to show what would have happened if we had continued to play the game for 10 more rounds. Why would the graph look like this?
9. What was the carrying capacity of the population of deer in Part I? How do you know?
10. This game was used to model how populations change. What parts of the game were not as they are in nature? (What parts were not real? What did we assume?)

#### Part II: Deer, Resources, and Wolves

11. Which trophic level do the wolves represent?
12. What kind of a limiting factor (density-dependent or density independent) do the wolves represent? Explain.
13. How do predator and prey populations affect each other?
14. Why does one population increase AFTER the other population increases?
15. Is the carrying capacity different in Part II than it is in Part I? If so, why? If not, why is the maximum number of deer in the population lower in Part I than in Part II?

P5

# PART 1 (deer and resources)

approximately 10 rounds

ROUND (generation)	# of DEER (population)
0	8
1	14
2	13
3	18
4	10
5	18
6	10
7	18
8	12
9	10
10	10

01

# PART 2 (deer, resources, and predators)

approximately 10 rounds

ROUND (generation)	# of DEER (population)	# of WOLVES (population)
0	8	2
1	10	4
2	6	6
3	2	12
4	2	2
5	0	4
6	6	4
7	8	4
8	10	6
9	10	6
10	↓	8