

R-6.7 Would you use the adjacency list structure or the adjacency matrix structure in each of the following cases? Justify your choice.

- The graph has 10,000 vertices and 20,000 edges, and it is important to use as little space as possible.
- The graph has 10,000 vertices and 20,000,000 edges, and it is important to use as little space as possible.
- You need to answer the query areAdjacent as fast as possible, no matter how much space you use.

Asymptotically,

	Adjacency List	Adjacency Matrix
Space	$N + M$	N^2
areAdjacent	$\text{Min}(\deg(v), \deg(w))$	1

N = number of vertices

M = number of edges

$\deg(v)$ = degree of a vertex v , i.e. number of graph edges which touch ' v '

a.) $N = 10000$ and $M = 20000$

$$\text{Space for Adjacency List} = 10000 + 20000 = 30000 = 3 \times 10^4$$

$$\text{Space for Adjacency Matrix} = 10000^2 = 100000000 = 10^8$$

$$3 \times 10^4 < 10^8$$

Therefore Adjacency List will consume as little space as possible

b.) $N = 10000$ and $M = 20000000$

$$\text{Space for Adjacency List} = 10000 + 20000000 = 20010000 = 2 \times (10^7) + 10^4$$

$$\text{Space for Adjacency Matrix} = 10000^2 = 100000000 = 10^8 = 10 \times (10^7)$$

$$2 \times (10^7) + 10^4 < 10^8$$

Therefore Adjacency List will consume as little space as possible

c.) given "no matter how much space you use", means no constraint on space but require least time

Then Adjacency Matrix will answer the query "areAdjacent" in $O(1)$

