1. How did we show that the Gale-Shapley algorithm always returns a stable matching?

A. By arguing.
B. By creating a chain of deductive reasoning.
C. By creating a proof by contradiction.
D. By creating a proof by induction.
E. We didn’t need to since we had already created a proof by contradiction that it returns a perfect matching.
2. Suppose we are looking at a matching of men and women returned by the Gale-Shapely algorithm. The matching contains \((m, w)\) and \((m', w')\).

An instability occurs in a matching when:

A. \(m\) prefers \(w'\) to \(w\)
B. \(m\) prefers \(w'\) to \(w\) and \(w\) prefers \(m'\) to \(m\)
C. \(m\) prefers \(w'\) to \(w\) and \(w'\) prefers \(m\) to \(m'\)
D. \(m'\) prefers \(w\) to \(w'\) and \(w\) prefers \(m\) to \(m'\)
E. \(w'\) prefers \(m\) to \(m'\)
3. If we are not limited to using the Gale-Shapley algorithm, is it possible for there to be more than one stable matching?

A. Yes, depending on the preference lists.

B. No, no matter whether you use the G-S algorithm or not, there is always only 1 stable matching.
4. Can the Gale-Shapley algorithm return different stable matchings if the input remains the same? (Input includes both preference lists and who is doing the asking.)

A. Yes, depending on the preference lists.
B. No, G-S always returns the same stable matching for a given input.
5. How can we make the Gale-Shapley algorithm return a different stable matching?

A. We can’t. No matter which group is doing the asking we always get the same stable matching from this algorithm.

B. Switch the group doing the asking.

C. Run the algorithm multiple times since it is non-deterministic.
6. Interval scheduling is an example of a class of problems that can be solved using:

A. augmentation algorithms (maximum flow)
B. divide and conquer algorithms
C. dynamic programming
D. greedy algorithms
E. no known efficient algorithms exist for this class of problems
7. **Weighted** interval scheduling is an example of a class of problems that can be solved using:

A. augmentation algorithms (maximum flow)
B. divide and conquer algorithms
C. dynamic programming
D. greedy algorithms
E. no known efficient algorithms exist for this class of problems
8. Bipartite matching is an example of a class of problems that can be solved using:

A. augmentation algorithms (maximum flow)
B. divide and conquer algorithms
C. dynamic programming
D. greedy algorithms
E. no known efficient algorithms exist for this class of problems
9. Finding an independent set is an example of a class of problems that can be solved using:

A. augmentation algorithms (maximum flow)
B. divide and conquer algorithms
C. dynamic programming
D. greedy algorithms
E. no known efficient algorithms exist for this class of problems
10. Finding an independent set is a generalization of what other kinds of problems?

A. interval scheduling
B. bipartite matching
C. weighted interval scheduling and bipartite matching
D. interval scheduling and bipartite matching
E. none of them
Answers:
1. C
2. C
3. A
4. B
5. B
6. D
7. C
8. A
9. E
10. D