cmput 304 2024 study questions 5 (with hints)

1. a) Describe a brute force algorithm that takes a graph as input and returns a largest clique.

   b) Describe a polytime algorithm that takes a graph and a node subset as input and reports whether the node subset is a clique.

   c) Explain how b) is used in proving that the k-clique problem is in the class NP.

2. *k-clique* is this problem:

   instance: a graph G and an integer k

   query: does G have a clique of size k?

   a) Define the problem *k-independent set*.

   b) Explain why the brute force method for solving k-clique runs in polytime if k is constant but in $\Omega(2^n/\sqrt{n})$ time when $k = n/2$.

   c) Give a polytime answer-preserving transformation T from k-clique to k-independent set (so T takes as input an instance of k-clique and gives as output an instance of k-independent set). Prove that T is polytime. Prove that T is answer-preserving.

3. a) Define NP-complete.

   b) So far in the lectures we have seen that these problems are NP-complete: conjunctive normal form sat (cnf-sat); 3-cnf-sat, also called 3-sat; k-clique; k-independent set.

   For each of these problems, explain briefly how we know that the problem is in the class NP-complete.

   c) Does there exist a polytime answer-preserving transformation from sat to k-independent set? If yes, explain briefly how you know this. If no, explain briefly why not.

   d) Is there a polytime answer-preserving transformation from k-independent set to sat? If yes, explain briefly how you know this. If no, explain briefly why not.

4. From the lectures, let T be the transformation from cnf-sat to 3-sat. For the cnf-sat formula f represented below, give the corresponding 3-sat formula $T(f)$.

```
f = [-1 3 5 6] [2 3 -4 -5 -6] [1 -2 3 -4 5 6]
```

5. From the lectures, let T be the transformation from cnf-sat to 3-sat.

   Recall: this cnf-sat formula s ...

```
[  1  2  3  -4  -6 ]
[  1  2 -3  -5     ]
[  1 -2  3  -5  -6 ]
[  1  3  4         ]
[  2 -3  4 -6      ]
```

   ... is transformed to this 3-sat formula t:

```
[   1   2  10 ]
[ -10   3  11 ]
[ -11  -4  -6 ]
[   1   2  12 ]
[ -12  -3  -5 ]
[  1   -2  13 ]
[ -13   3  14 ]
[ -14  -5  -6 ]
[   1   3   4 ]
[   2  -3  15 ]
[ -15   4  -6 ]
```

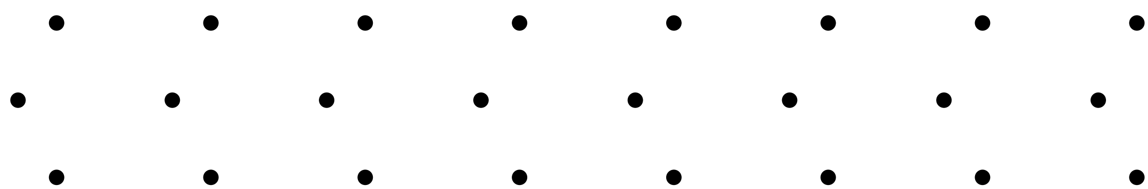
   Let $f()$ be an assignment of boolean values to variables such that s is unsatisfied, i.e. $f(s) =$ False. Prove that t is unsatisfied, i.e. $f(s) =$ False.

   Hint: break this problem into five subproblems, corresponding to the five clauses of f. Pick your favorite subproblem, and answer that subproblem (and don't pick the 4th subproblem, that's too easy).

6. Let T be the transformation we saw in the lectures from 3-sat to k-independent set.

   a) For formula f below, label the nodes below and draw the edges of the graph $T(f)$.

$$\begin{aligned} f = & (\neg x_1 \lor x_2 \lor x_3) \land (x_2 \lor x_3 \lor \neg x_4) \land (x_1 \lor x_3 \lor x_4) \land (\neg x_1 \lor \neg x_2 \lor \neg x_4) \land \\ & (x_1 \lor \neg x_2 \lor \neg x_3) \land (x_1 \lor x_2 \lor \neg x_3) \land (\neg x_1 \lor \neg x_2 \lor x_4) \land (\neg x_1 \lor x_2 \lor \neg x_3) \end{aligned}$$



   b) If $T(f)$ has an 8-independent set, circle each node of such a set. If it does not, explain briefly.

   c) If f has a satisfying assignment, give it below. If it does not, explain briefly.

```
variable      1  2  3  4
truth value  __ __ __ __
```

7. 2-coloring is this problem.

   instance: a graph G

   query: is there an assignment of at most 2 colors to nodes of G, such that pair of adjacent nodes have different colors?

   a) Give a graph with 3 nodes that is not 2-colorable, and a graph with 6 nodes and six edges that is 2-colorable.

   b) Give a polytime answer-preserving transformation T from 2-coloring to sat. Prove that T is polytime. Prove that T is answer-preserving.

   c) Does b) imply that 2-coloring is in the class NP-complete? Explain briefly.

8. Is there a polytime answer-preserving transformation from k-independent set to satisfiability? If yes, explain briefly how you know this. If no, explain briefly why not (or why this is unlikely to be true).

9. See also quiz 5 from 2023 (and the solutions).

hints

1. a) for each vertex subset, check whether it is a clique. track largest-so-far b) for each pair of nodes in the subset, check whether they are adjacent. if any such pair is non-adjacent, it is not a clique c) for any yes-instance, there exists a k-clique, so find one (take as long as you want, say exponential time) and then use it for verification (can be checked in polytime)

2. a) replace the word *clique* with *independent set* b) the number of k-subsets of an n-set is n choose k, which is in $O(n^k)$, which is polynomial in n when k is constant, but in $\Omega(2^n/\sqrt{n})$ when $k = n/2$ (see the webnotes) c) given a graph G, replace it with its complement H (replace each adjacent pair of nodes with a non-adjacent pair, and vice versa).

3. a) see webnotes and lecture b) cnf-sat is NP-C by Cook's theorem. in class we saw that cnf-sat transforms to 3-sat, and that 3-sat transforms to k-clique, and in the previous question that k-clique transforms to k-independent set. c) yes: by b) we know that k-ind. set is NP-complete, so any problem in NP can be transformed into it, so sat (which is in NP) can be transformed into it (and here, everytime we say *transformed* we mean *answer-preserving polytime transformed*).

4. covered in the lectures

5. covered in the lectures

6. a) triangle. 6-cycle b) one boolean variable for each node. two colors, taupe (corresponding to value true) and fuscia (corresponding to value false). for each edge, say from node j to k, have the clause (v_j and not v_k) or (v_k and not v_j) c) no: the transformation is going in the wrong direction for that. we know from Cook's theorem that every problem in NP (e.g. 2-coloring) can be polytime transformed into sat, so seeing this transformation does not surprise us.

7. yes: every problem in NP (e.g. ind. set) transforms in any NP-complete problem (e.g. sat)

8. a) see the lecture bi) the graph partitions into m triangles. a triangle is a clique, so any independent set includes at most one node from each triangle, so any independent set has at most m nodes (otherwise, there would be some triangle that included at least two nodes from the IS, a contradiction) bii) see the lecture