Module 7 Graded Quiz

Due Nov 19, 2021 at 11:59pm **Points** 15
Questions 15
Available Nov 6, 2021 at 12am - Jan 21 at 11:59pm 3 months
Time Limit 120 Minutes

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	29 minutes	15 out of 15

Score for this quiz: **15** out of 15
Submitted Nov 19, 2021 at 6:35pm
This attempt took 29 minutes.

Question 1

1 / 1 pts

Which of the following problems does not likely have a polynomial-time algorithm? Select all that apply.

- [ ] Shortest Path
- [ ] Minimum Cut
- Correct! [x] Knapsack
- Correct! [x] Vertex Cover

Question 2

1 / 1 pts

What is a decision problem?

- A problem that is in P or NP.

Correct!

- A problem in which the answer is always “yes” or “no”.
- A problem that requires any algorithm for it to return a result that is not a boolean value.
- A problem that requires any algorithm for it to make a decision.

Question 3

1 / 1 pts

Why, informally, is P a subset of NP?

Correct!

- Because the certificate could be the algorithm itself, and the certifier could be the execution of the algorithm.
- Because any polynomial-time algorithm obviously runs in polynomial-time.
- Because it follows directly from the definition.

Question 4

1 / 1 pts

Which of the following is a valid statement about problem Y for when X $\leq_P$ Y and X is guaranteed to be polynomial-time solvable?

Correct!

- Nothing can be guaranteed about Y.
- Y is not in P.

- Y is in P.
- Y is equivalent to X.

Question 5

1 / 1 pts

If problems X and Y are equivalent (i.e. X reduces in poly-time to Y and Y reduces in poly-time to X), what can we conclude about their run times?

- They are within a constant factor.
- Correct! They are at most a polynomial factor different.
- They are at most an exponential factor different.
- They are identical.

Question 6

1 / 1 pts

Which of the following is a valid set of constraints on the inputs to the VERTEX-COVER problem?

- Correct! A graph, and a nonnegative integer k.
- A directed graph, and an arbitrary integer k.
- A directed graph, and a nonnegative integer k.
- A graph, and an arbitrary integer k.

Question 7

1 / 1 pts

In the SAT problem, what does “conjunctive normal form” mean?

Correct!

- (selected) The formula is a conjunction of clauses, which are disjunctions of literals.
- The formula is a disjunction of clauses, which are disjunctions of literals.
- The formula is a disjunction of clauses, which are disjunctions of literals.
- The formula is a conjunction of clauses, which are conjunctions of literals.

Question 8

1 / 1 pts

Consider the following algorithm for VERTEX-COVER: given the graph G and integer k, test all $O(n^k)$ sets of k vertices, and for each determine if it is a vertex cover. Is this a polynomial-time algorithm? Why/why not?

- No, because no polynomial time algorithm exists for VERTEX-COVER.
- Yes, because n^k is a polynomial.

Correct!

- (selected) No, since k is part of the input and it is not constrained to be a constant
- Yes, because there are polynomially many subsets of the vertices.

Question 9

1 / 1 pts

What was the “clause gadget” in the 3-SAT to INDEPENDENT-SET reduction?

Correct!

- (selected) A triangle (cycle of length 3).
- A path of length 3.
- A complete graph of size O(n) (n is the number of variables).
- A single vertex with no edges.

Question 10

1 / 1 pts

Why is the $\leq_P$ relation transitive? (i.e., X $\leq_P$ Y and Y $\leq_P$ Z implies that X $\leq_P$ Z)

- Because a polynomial plus another is also a polynomial.
- None of the other answers is correct.
- Because a polynomial times another is also a polynomial.

Correct!

- (selected) Because the reductions can be composed together into a polynomial time reduction from problem X to problem Z.

Question 11

1 / 1 pts

What if in the 3-SAT problem you were allowed to have at most 3 literals per clause (the original problem specified exactly 3)? Please mark which of the following are correct.

- The modified version of the 3-SAT problem is the same as the 2-SAT problem.
- The modified version of the 3-SAT problem is in P, and 3-SAT is NP complete.
- The modified version of the 3-SAT problem is in P, and 3-SAT is not in P.

Correct!

- The modified version of the 3-SAT problem is polynomially equivalent to the original version of 3-SAT.

Question 12

1 / 1 pts

Which of the following is not an example of a known NP-complete problem?

- KNAPSACK

Correct!

- PRIMES
- TRAVELING SALESMAN PROBLEM
- SUBSET-SUM

Question 13

1 / 1 pts

My friend is trying to show X$\leq_P$ Y for problems X,Y. He has given a polynomial time function that transforms an instance I of X into an instance poly(I) of Y. Lastly, he has shown that if the decision is yes for poly(I), then the decision is always yes for I. Is my friend's proof complete?

- Yes, this completes all the steps to show a polynomial time reduction from X to Y since yes decisions of Y are mapped to no decisions for X.
- No, your friend must show a decision of no for poly(I) corresponds to a decision of yes for I.
- Yes, this completes all the steps to show a polynomial time reduction from X to Y since yes decisions of Y are mapped to yes decisions for X.

Correct!

- (selected) No, your friend must show a decision of no for poly(I) corresponds to a decision of no for I.

Question 14

1 / 1 pts

The statements below concern the classes P and NP. Mark all correct statements.

Correct!

- [x] P and NP consist only of decision problems.
- [ ] The version of the Knapsack problem wherein one must find a subset of maximum profit subject to weight constraints is in NP.

Correct!

- [x] $|P| \leq |NP|$

Correct!

- [x] The version of the Traveling Salesman problem wherein one must determine whether a tour of weight at least W exists is in NP.

Question 15

1 / 1 pts

If an instance I of a problem X undergoes a transformation to an instance I' of a problem Y, and this transformation terminates in time polynomial in the size of I , then must the size of I' be polynomial in the size of I?

- ( ) No, problem X might be NP-complete, and thus any transformation of I will produce a result exponential in the size of I.

Correct!

- (•) Yes, a Turing machine that runs for $|I|^c$ transitions for some constant c can only access at most $|I|^c$ tape cells.
- ( ) No, even it the transformation terminates in polynomial time, it could have used an exponential amount of space in its computation.
- ( ) Yes, since the size of I' must be equal to the size of I.

Quiz Score: **15** out of 15