Problem

Show that any PSPACE-hard language is also NP-hard.

Step-by-step solution

Step 1 of 2

PSPACE –completeness: A language B is PSPACE – complete if it satisfies two conditions

1. B exists in PSPACE, and

2. Every A is PSPACE is polynomial time reducible to B.

If B satisfies condition 2, we say that it is PSPACE – hard

Comment

Step 2 of 2

NP – Completeness : A language B is *NP* – complete if it satisfies two conditions.

1. B is in *NP*, and

2. Every A is *NP* is polynomial time reducible to B

If B satisfies condition 2, we say that it is NP – hard.

Now we have to show that any PSPACE – hard language is NP – hard.

We know that NP is subset of PSPACE.

Therefore any string outside of PSPACE is also outside of NP.

In any problem from NP will reduced to PSPACE _ hard language.

- We know that "SAT = { <Φ>| is a satisfiable Boolean formula}."
- Also we know that "TQBF ={ < Φ>| is a true fully quantified Boolean formula}."
- We know that $SAT \in$ NP-complete
- Since any SAT problem can be reduced to a TQBF problem by simply adding "there exist x_n" to the front of SAT expression for each variable x_n.

So SAT problem can be solved using TQBF algorithm

- But *TQBF* problem is reduced to any *PSPACE* – hard problem. Because as we know that *TQBF* is *PSPACE* –complete.
- Thus *SAT* is reducing to *PSPACE* – hard, that means *NP* – hard problem is reduced to *PSPACE* – hard.

Thus any *PSPACE* – hard is also *NP* – hard.

Comment