Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	<TO BE FILLED BY STUDENT>

TUTORIAL SESSION 23:

PSPACE

Concept Building

PSPACE is a complexity class that represents the set of decision problems that can be solved by a Turing machine using a polynomial amount of space. This means that the amount of memory used by the algorithm grows polynomially with the size of the input, regardless of the time it takes to compute the solution.

Key Characteristics of PSPACE

1. **Definition**: A decision problem is in PSPACE if there exists an algorithm that can solve the problem using a polynomial amount of memory space. The time complexity can be exponential or even worse, but the space used must be polynomial.
2. **Relation to Other Complexity Classes**:
   - **P**: The class of problems that can be solved in polynomial time. It is a subset of PSPACE, meaning every problem in P can also be solved in PSPACE.
   - **NP**: The class of problems for which a solution can be verified in polynomial time. It is not known whether NP is a subset of PSPACE, but it is widely believed that PSPACE contains NP.
   - **PSPACE-complete**: A subset of PSPACE that contains the hardest problems in PSPACE. If any PSPACE-complete problem can be solved in polynomial time, then all problems in PSPACE can also be solved in polynomial time.
3. **Examples of PSPACE Problems**:
   - **Quantified Boolean Formula (QBF)**: This problem involves determining the truth of a quantified Boolean formula and is PSPACE-complete.
   - **Generalized Chess**: Determining the winner of a game of chess given any arbitrary position is PSPACE-complete.
   - **Tiling Problem**: The problem of determining whether a given set of tiles can tile a given area is PSPACE-complete.
4. **Space Complexity**: The space complexity of an algorithm is defined as the amount of memory it requires to solve a problem as a function of the input size. For example, an algorithm that uses space proportional to the square of the input size would have a space complexity of $O(n^2)$.
5. **PSPACE vs. Exponential Time**: While PSPACE allows for polynomial space usage, it does not limit the time complexity. Problems in PSPACE can take exponential time to solve, but they will not exceed polynomial space usage.
6. **Containment Relationships**:
   - $P \subseteq NP \subseteq PSPACE$

- PSPACE is believed to be strictly larger than NPNP, meaning there are problems in PSPACE that are not in NP.

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	<TO BE FILLED BY STUDENT>

Pre-Tutorial (To be completed by student before attending tutorial session)

1. Give an example of PSACE-Complete problem. Mention the reason for the same.

Solution:

Example: Quantified Boolean Formula (QBF)

Reason:

- Formula: $\exists x \, \forall y \, (x \lor \neg y)$.
- QBF uses polynomial space.
- Every PSPACE problem reduces to QBF in polynomial time.

2. What is the significance of Savitch's theorem in the context of PSPACE?

Solution:

Significance of Savitch's Theorem:

- **Formula**: $\mathrm{NPSPACE} = \mathrm{PSPACE}$
- **Implication**: Deterministic and nondeterministic polynomial space are equivalent.

3. If a polynomial-space algorithm is found for one PSPACE-complete problem, what can be concluded?

Solution:

Solution:
If a polynomial-space algorithm is found for one PSPACE-complete problem, it implies that **all PSPACE-complete problems** and **all problems in PSPACE** can be solved in polynomial space, as they are all reducible to each other.

IN-TUTORIAL (To be carried out in presence of faculty in classroom)

1. Give problems is known to be in PSPACE but not known to be PSPACE-complete?

Solution:

Problem: **Tarski's Circle-Squaring Problem**

Reason:

- Known to be in **PSPACE**.
- Not proven **PSPACE-complete** or **outside** PSPACE-complete boundaries yet.

2. What are the closure properties of PSPACE

Solution:

PSPACE Closure Properties:

1. **Complement**: Closed.
2. **Union**: Closed.
3. **Intersection**: Closed.
4. **Concatenation**: Closed.
5. **Polynomial-time reduction**: Closed.

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	<TO BE FILLED BY STUDENT>

3. Can you judge all PSPACE problems can be reduced to PSPACE Complete Problems

Solution:

Yes, all **PSPACE problems** can be reduced to **PSPACE-complete problems**. This is because **PSPACE-complete** problems are the hardest problems in PSPACE, and any problem in PSPACE can be polynomial-time reduced to a PSPACE-complete problem.

4. How can we measure the Polynomial Space.

Solution:

Polynomial space is measured by the memory usage $O(n^k)$, where k is a constant.

Post-Tutorial (To be carried out by student after attending tutorial session)

1. Write the relationship between PSPACE and EXPSPACE.

Solution:

Relationship between PSPACE and EXPSPACE:

- **PSPACE**: Solvable in polynomial space, $O(n^k)$.
- **EXPSPACE**: Solvable in exponential space, $O(2^{n^k})$.

Key relation:

$$\mathrm{PSPACE} \subset \mathrm{EXPSPACE}.$$

PSPACE problems can be solved in EXPSPACE, but not all EXPSPACE problems are in PSPACE.

2. Explain the PSPACE with example.

Solution:

PSPACE: Problems solvable with polynomial space.

Example: **TQBF (True Quantified Boolean Formula)**

- **Input**: ∃x ∀y (x ∨ ¬y)
- **Task**: Determine if the formula is true.
- **Explanation**: Solvable with polynomial space, but may require exponential time.

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	<TO BE FILLED BY STUDENT>

Viva – Questions

1. How does PSPACE relate to the classes P and NP?

Solution:

PSPACE contains both P and NP, as they use polynomial space.

2. What is the significance of PSPACE-complete problems?

Solution:

PSPACE-complete problems are the hardest problems in PSPACE, capturing complexity.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation Marks Secured: out of **50** Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation: