## 1. Class: `Tree`

- **Attributes**:
  - `treeSpecies`: Species of the tree (e.g., Maple, Fir, Birch).
  - `yearPlanted`: Year the tree was planted.
  - `treeHeight`: Height of the tree.
  - `treeGrowth`: Growth rate of the tree.
- **Methods**:
  - Constructors to initialize a tree with provided attributes.
  - Getters and setters for each attribute to access and modify the tree's properties.
  - `toString()` method to provide a string representation of the tree.
  - `generateRandomTree()` method to generate a random tree with random attributes.
- **Responsibilities**:
  - Represent individual trees within a forest.
  - Allow retrieval and modification of tree attributes.
  - Provide methods for generating random trees.

## 2. Class: `Forest`

- **Attributes**:
  - `name`: Name of the forest.
  - `trees`:  Array List of trees in the forest.
- **Methods**:
  - Constructor to initialize a forest with a name and an empty list of trees.
  - `printForest()` method to print details of each tree in the forest.
  - `addRandomTree()` method to add a randomly generated tree to the forest.
  - `cutTree()` method to cut down a tree by index.
  - `simulateYearlyGrowth()` method to simulate yearly growth for all trees in the forest.
  - `reapTrees()` method to remove trees exceeding a specified height and replace them with new random trees.
  - `saveForestToFile()` method to serialize the forest object and save it to a file.
  - `readForestFromCSV()` method to read forest data from a CSV file and populate the forest with trees.
  - `loadForestFromFile()` method to deserialize a forest object from a file.

- **Responsibilities**:
    - Manage forests and their associated trees.
    - Perform operations such as adding, cutting, and simulating growth of trees.
    - Handle serialization and deserialization of forest data.

## 3. Class: `ForestSimulation`

- **Attributes**:
    - `forestList`: Array List of forests.
    - `currentForestIndex`: Index of the current forest being simulated.
- **Methods**:
    - `main()` method to drive the simulation by displaying a menu and executing user-selected operations.
    - `initializeForestsFromCommandLine()` method to initialize forests from command-line arguments.
    - `displayMenu()` method to display a menu of options for interacting with the forest.
- **Responsibilities**:
    - Control the simulation flow.
    - Initialize forests from command-line arguments.
    - Provide a user interface for interacting with forests.