```java
//Harini.R 192324108
//JFo section-8
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
import java.util.Scanner;

class Team {
    private String name;
    private int wins;
    private int losses;
    private int ties;
    private int goalsScored;
    private int goalsAllowed;
    public Team(String name) {
        this.name = name;
        this.wins = 0;
        this.losses = 0;
        this.ties = 0;
        this.goalsScored = 0;
        this.goalsAllowed = 0;
    }
    public String getName() { return name; }
    public int getWins() { return wins; }
    public int getLosses() { return losses; }
    public int getTies() { return ties; }
    public int getGoalsScored() { return goalsScored; }
    public int getGoalsAllowed() { return goalsAllowed; }

    public void addWin() { wins++; }
    public void addLoss() { losses++; }
    public void addTie() { ties++; }
    public void addGoalsScored(int goals) { goalsScored += goals; }
    public void addGoalsAllowed(int goals) { goalsAllowed += goals; }

    public void printStats() {
```

```
        System.out.println(name);
        System.out.println("Wins: " + wins + ", Losses: " + losses + ", Ties: " + ties);
        System.out.println("Goals Scored: " + goalsScored + ", Goals Allowed: " + goalsAllowed);
    }
}

class Game {
    private int gameId;
    private Team homeTeam;
    private Team awayTeam;
    private int temperature;
    private int homeTeamScore;
    private int awayTeamScore;

    public Game(int gameId, Team homeTeam, Team awayTeam, int temperature) {
        this.gameId = gameId;
        this.homeTeam = homeTeam;
        this.awayTeam = awayTeam;
        this.temperature = temperature;
        this.homeTeamScore = generateScore(temperature);
        this.awayTeamScore = generateScore(temperature);
    }

    private int generateScore(int temperature) {
        return (int) (Math.random() * (temperature / 10)); // Example logic
    }

    public void updateTeamStats() {
        homeTeam.addGoalsScored(homeTeamScore);
        homeTeam.addGoalsAllowed(awayTeamScore);
        awayTeam.addGoalsScored(awayTeamScore);
        awayTeam.addGoalsAllowed(homeTeamScore);

        if (homeTeamScore > awayTeamScore) {
            homeTeam.addWin();
            awayTeam.addLoss();
```

```
        } else if (homeTeamScore < awayTeamScore) {
            awayTeam.addWin();
            homeTeam.addLoss();
        } else {
            homeTeam.addTie();
            awayTeam.addTie();
        }
    }

    public void printGameDetails() {
        System.out.println("Game #" + gameId);
        System.out.println("Temperature: " + temperature);
        System.out.println("Home Team: " + homeTeam.getName() + ", Score: " + homeTeamScore);
        System.out.println("Away Team: " + awayTeam.getName() + ", Score: " + awayTeamScore);
    }

    public int getTemperature() { return temperature; }
}

class Scheduler {
    private List<Team> teams;
    private List<Game> games;
    private int gameIdCounter;
    private int consecutiveFreezingWeeks;

    public Scheduler(List<Team> teams) {
        this.teams = teams;
        this.games = new ArrayList<>();
        this.gameIdCounter = 1;
        this.consecutiveFreezingWeeks = 0;
    }

    public void scheduleWeek(int temperature) {
        if (temperature < 32) {
            System.out.println("Too cold to play.");
            consecutiveFreezingWeeks++;
```

```
109             if (consecutiveFreezingWeeks == 3) {
110                 System.out.println("Season is over due to cold weather.");
111                 printSeasonResults();
112                 System.exit(0);
113             }
114         } else {
115             consecutiveFreezingWeeks = 0;
116             playGames(temperature);
117         }
118     }
119
120     private void playGames(int temperature) {
121         Collections.shuffle(teams);
122
123         for (int i = 0; i < teams.size(); i += 2) {
124             Team homeTeam = teams.get(i);
125             Team awayTeam = teams.get(i + 1);
126             Game game = new Game(gameIdCounter++, homeTeam, awayTeam, temperature);
127             games.add(game);
128             game.updateTeamStats();
129             game.printGameDetails();
130         }
131     }
132
133     public void printSeasonResults() {
134         System.out.println("*********RESULTS*********");
135
136         for (Team team : teams) {
137             team.printStats();
138             System.out.println();
139         }
140
141         int maxTemp = Integer.MIN_VALUE;
142         int totalTemp = 0;
143         for (Game game : games) {
144             maxTemp = Math.max(maxTemp, game.getTemperature());
```

```
            totalTemp += game.getTemperature();
        }

        System.out.println("Hottest Temp: " + maxTemp);
        System.out.println("Average Temp: " + (games.size() > 0 ? (totalTemp / games.size()) : 0));
    }
}

public class SoccerLeagueTest {
    public static void main(String[] args) {
        List<Team> teams = new ArrayList<>();
        teams.add(new Team("Team 1"));
        teams.add(new Team("Team 2"));
        teams.add(new Team("Team 3"));
        teams.add(new Team("Team 4"));

        Scheduler scheduler = new Scheduler(teams);
        Scanner scanner = new Scanner(System.in);

        while (true) {
            System.out.print("Enter the temperature for this week's games: ");
            int temperature = scanner.nextInt();
            scheduler.scheduleWeek(temperature);
        }
    }
}
```

```
Enter the temperature for this week's games: 90
Game #1
Temperature: 90
Home Team: Team 1, Score: 2
Away Team: Team 3, Score: 1
Game #2
Temperature: 90
Home Team: Team 2, Score: 6
Away Team: Team 4, Score: 3
Enter the temperature for this week's games: 35
Game #3
Temperature: 35
Home Team: Team 2, Score: 2
Away Team: Team 1, Score: 0
Game #4
Temperature: 35
Home Team: Team 3, Score: 2
Away Team: Team 4, Score: 0
Enter the temperature for this week's games: 28
Too cold to play.
Enter the temperature for this week's games: 30
Too cold to play.
Enter the temperature for this week's games: 31
Too cold to play.
Season is over due to cold weather.
*********RESULTS*********
Team 2
Wins: 2, Losses: 0, Ties: 0
Goals Scored: 8, Goals Allowed: 3

Team 1
Wins: 1, Losses: 1, Ties: 0
Goals Scored: 2, Goals Allowed: 3

Team 3
Wins: 1, Losses: 1, Ties: 0
Goals Scored: 3, Goals Allowed: 2

Team 4
Wins: 0, Losses: 2, Ties: 0
Goals Scored: 3, Goals Allowed: 8

Hottest Temp: 90
Average Temp: 62
```