SAT: Property-based Testing

1. Write property-based tests for the following procedure:

```java
public double calculate(double income) {
    if (0 <= income && income < 22100) {
        return 0.15 * income;
    } else if (22100 <= income && income < 53500) {
        return 3315 + 0.28 * (income - 22100);
    } else if (53500 <= income && income < 115000) {
        return 12107 + 0.31 * (income - 53500);
    } else if (115000 <= income && income < 250000) {
        return 31172 + 0.36 * (income - 115000);
    } else if (250000 <= income) {
        return 79772 + 0.396 * (income - 250000);
    }
    return -1;
}
```

2. Write property-based tests for the following procedure:

```java
/**
 * This method predicts whether it is summer.
 * If at least 75\% of the temperature values provided are 20 degrees
 * or above, it is summer. Otherwise, it is not summer.
 * @param temperatures The list of temperature values
 * @return the probability of it being summer
 */
public static boolean isItSummer(List<Double> temperatures) {
    int count20OrAbove = 0;
    for (Double temp : temperatures) {
        if (temp >= 20) {
            count20OrAbove++;
        }
    }
    return count20OrAbove >= temperatures.size() * 0.75f;
}
```

3. Write property-based tests for the following procedure:

```
enum GameResult {
    X_WON,
    O_WON,
    DRAW
}
public static GameResult calculateResult(String board) {
    // Check if X won
    boolean xWon = checkWin(board, 'X');
    // Check if O won
    boolean oWon = checkWin(board, 'O');
    // Check if both have 4 in a row (this should not happen)
    if (xWon && oWon) {
        throw new IllegalArgumentException();
    }
    if (xWon) {
        return GameResult.X_WON;
    }
    if (oWon) {
        return GameResult.O_WON;
    }
    return GameResult.DRAW;
}
private static boolean checkWin(String board, char player) {
    int playerCharacters = 0;
    for (int i = 0; i < board.length(); i++) {
        if (board.charAt(i) == player) {
            playerCharacters++;
            if (playerCharacters == 4) {
                return true;
            }
        } else {
            playerCharacters = 0;
        }
    }
    return false;
}
```