The parallel program was tested under many different parameters, specifically the size of the matrix. It was tested on randomly generated matrices of sizes N = 64; N = 256; N = 502; N = 1024; and N = 4096. The triangular matrix was printed correctly in all of the tests. A checkSolution method was created to verify the solutions of each test

```
public static boolean checkSolution(double[][] matrix, double[] sol) {
    int n = matrix[0].length;
    for(int x = 0; x < matrix.length; x++){
        double sum = 0;
        for(int y = 0; y < n - 1; y++){
            sum += matrix[x][y]*sol[y];
        }

        if(sum - matrix[x][n - 1] <= -0.01 || sum - matrix[x][n - 1] >= 0.01){
            return false;
        }
    }
    return true;
}
```

The method takes the sum of each element in a row multiplied by its corresponding solution to make sure it matches the end value. This is done for each row, and if a single row fails the test the solution is rejected. There is a small leeway given with the calculation due to the interference of minute decimals. As long as the sum is within 2 hundredths of the correct answer then it is accepted.