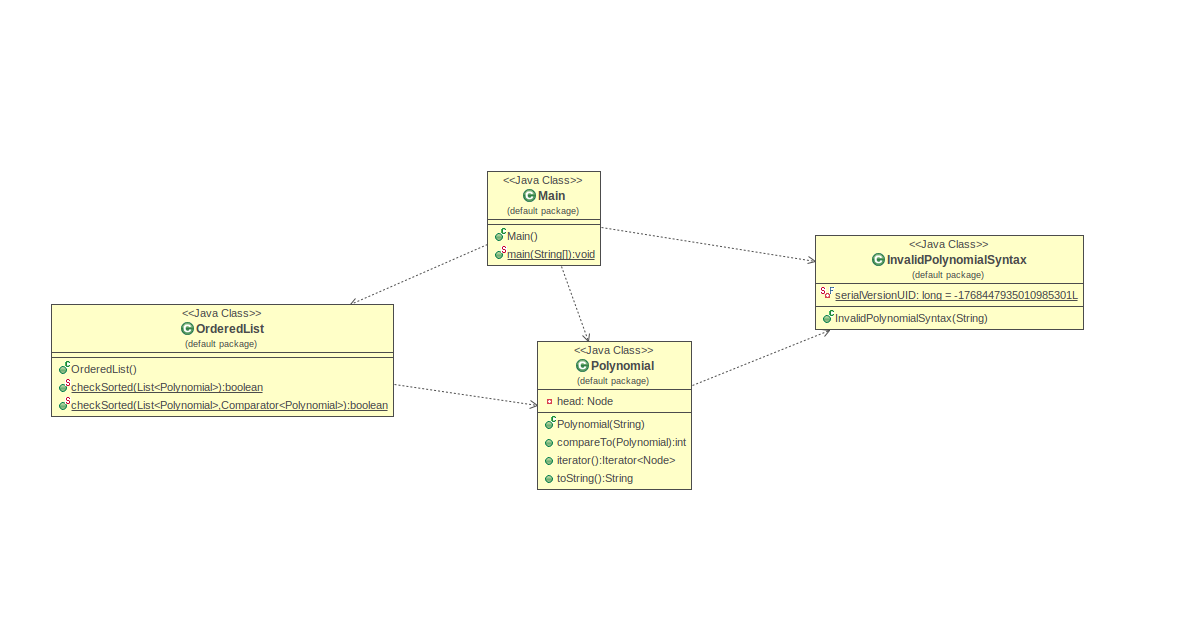
CMSC 350 Project 2

**Class Diagram**

****

**Test Cases 1: Test Cases include a file in both strong and weak sorted order**

Test Case 1a:

1. Create and save a text file (testcase1a.txt) that contains the following polynomials:

2 3 5 1

4 5 8 2

6 7 9 3

1. After launching the application, select the polynomial input file (testcase1a.txt) from the JFileChooser and press the save button.
2. The command line or console should display:

2.0x^3 + 5.0x

4.0x^5 + 8.0x^2

6.0x^7 + 9.0x^3

Polynomials are in strict ascending order:true

Polynomials are in weak ascending order:true

Test Case 1b:

1. Create and save a text file (testcase1b.txt) that contains the following polynomials:

3 2 6 1 4 0

5 3 4 2 2 1 5 0

10 4 6 3 5 2 3 1 1 0

1. After launching the application, select the polynomial input file (testcase1b.txt) from the JFileChooser and press the save button.
2. The command line or console should display:

3.0x^2 + 6.0x + 4.0

5.0x^3 + 4.0x^2 + 2.0x + 5.0

10.0x^4 + 6.0x^3 + 5.0x^2 + 3.0x + 1.0

Polynomials are in strict ascending order:true

Polynomials are in weak ascending order:true

**Test Cases 2: Test Cases include a file in weak but not strong sorted order**

Test Case 2a:

1. Create and save a text file (testcase2a.txt) that contains the following polynomials:

2 2 1 1 3 0

3 3 6 1 5 0

1 3 4 2 5 1 2 0

1. After launching the application, select the polynomial input file (testcase2a.txt) from the JFileChooser and press the save button.
2. The command line or console should display:

2.0x^2 + 1.0x + 3.0

3.0x^3 + 6.0x + 5.0

1.0x^3 + 4.0x^2 + 5.0x + 2.0

Polynomials are in strict ascending order:false

Polynomials are in weak ascending order:true

Test Case 2b:

1. Create and save a text file (testcase2b.txt) that contains the following polynomials:

1 3 2 1 4 0

5 4 6 1 5 0

2 4 3 3 5 1 2 0

1. After launching the application, select the polynomial input file (testcase2b.txt) from the JFileChooser and press the save button.
2. The command line or console should display:

1.0x^3 + 2.0x + 4.0

5.0x^4 + 6.0x + 5.0

2.0x^4 + 3.0x^3 + 5.0x + 2.0

Polynomials are in strict ascending order:false

Polynomials are in weak ascending order:true

**Test Cases 3: Test Cases include a file in neither strong nor weak sorted order**

Test Case 3a:

1. Create and save a text file (testcase3a.txt) that contains the following polynomials:

2 2 1 1 4 0

5 3 3 2 1 1 2 0

4 3 2 1 2 0

1. After launching the application, select the polynomial input file (testcase3a.txt) from the JFileChooser and press the save button.
2. The command line or console should display:

2.0x^2 + 1.0x + 4.0

5.0x^3 + 3.0x^2 + 1.0x + 2.0

4.0x^3 + 2.0x + 2.0

Polynomials are in strict ascending order:false

Polynomials are in weak ascending order:false

Test Case 3b:

1. Create and save a text file (testcase3b.txt) that contains the following polynomials:

1 3 3 2 5 0

4 4 3 3 7 2 4 0

3 4 2 2 1 0

1. After launching the application, select the polynomial input file (testcase3b.txt) from the JFileChooser and press the save button.
2. The command line or console should display:

1.0x^3 + 3.0x^2 + 5.0

4.0x^4 + 3.0x^3 + 7.0x^2 + 4.0

3.0x^4 + 2.0x^2 + 1.0

Polynomials are in strict ascending order:false

Polynomials are in weak ascending order:false

**Test Cases 4: Test Cases include files with both kinds of syntax errors**

Test Case 4a:

1. Create and save a text file (testcase4a.txt) that contains the following polynomials:

3 1 4 2 2 4

4

1. After launching the application, select the polynomial input file (testcase4a.txt) from the JFileChooser and press the save button.
2. The command line or console should display a JOptionPane window containing the error message “Invalid Polynomial.Invalid Polynomial.Exponents are not in sorted order” and another JOptionPane window containing the error message “Invalid Polynomial.1”.

The console should also display:

Polynomials are in strict ascending order:true

Polynomials are in weak ascending order:true

Test Case 4b:

1. Create and save a text file (testcase4b.txt) that contains the following polynomials:

2 1 8 5 6 2

10 k

1. After launching the application, select the polynomial input file (testcase4b.txt) from the JFileChooser and press the save button.
2. The command line or console should display a JOptionPane window containing the error message “Invalid Polynomial.Invalid Polynomial.Exponents are not in sorted order” and another JOptionPane window containing the error message ‘Invalid Polynomial.For input string: “k”’

The console should also display:

Polynomials are in strict ascending order:true

Polynomials are in weak ascending order:true

**Test Cases 5: Test Cases include a polynomial with exponents of 0, 1 and 2 or more**

Test Case 5a:

1. Create and save a text file (testcase5a.txt) that contains the following polynomials:

5 2 3 1 4 0

3 2 3 1 1 0

1. After launching the application, select the polynomial input file (testcase5a.txt) from the JFileChooser and press the save button.
2. The command line or console should display:

5.0x^2 + 3.0x + 4.0

3.0x^2 + 3.0x + 1.0

Polynomials are in strict ascending order:false

Polynomials are in weak ascending order:true

Test Case 5b:

1. Create and save a text file (testcase5b.txt) that contains the following polynomials:

5 2 7 1 2 0

3 3 4 2 1 1

1. After launching the application, select the polynomial input file (testcase5b.txt) from the JFileChooser and press the save button.
2. The command line or console should display:

5.0x^2 + 7.0x + 2.0

3.0x^3 + 4.0x^2 + 1.0x +

Polynomials are in strict ascending order:true

Polynomials are in weak ascending order:true

**Lessons Learned:**

From this project, I had the opportunity to work more with GUI. I learned how to utilize a JFileChooser to enable the user to effectively select a file for input. This project helped me gain experience and exposure with reading information from a file and effectively mixing data structures into the equation. I have better working knowledge of linked lists and nodes due to creating my own linked lists and nodes rather than using the predefined versions of those classes. This project was also useful in practicing and learning how to handle exceptions in case things to wrong in the user’s selected file. This project built on previous topics I learned in my other java experiences/exposures such as creating a custom named exception. It was interesting using linked lists and nodes to represent a polynomial in an input file and retrieving, storing, and utilizing that data.