AP Computer Science **Final Project**

Option 3: The Cebysev Polynomials

This Java project will generate the first 10 Cebysev polynomials using the Recurrence Formula:

$$T_{n+1}(x) = 2x(T_n(x)) - T_{n-1}(x)$$

The list of the first 10 Cebysev polynomials can be found below:

$T_0(x) = 1$
$T_1(x) = x$
$T_2(x) = 2x^2 - 1$
$T_3(x) = 4x^3 - 3x$
$T_4(x) = 8x^4 - 8x^2 + 1$
$T_5(x) = 16x^5 - 20x^3 + 5x$
$T_6(x) = 32x^6 - 48x^4 + 18x^2 - 1$
$T_7(x) = 64x^7 - 112x^5 + 56x^3 - 7x$
$T_8(x) = 128x^8 - 256x^6 + 160x^4 - 32x^2 + 1$
$T_9(x) = 256x^9 - 576x^7 + 432x^5 - 120x^3 + 9x$

Task #1: Create a Term class

This class should have an integer instance field for the coefficient and an integer instance field for the exponent. There should be an accessor method for each instance field. I would also suggest that you develop a method for multiplying 2 terms together. The header of that method might look like this:

```
public Term multiply(Term t)
```

You might also want a toString method so the project knows how you want a Term to look when it is printed. I would suggest that $- 20x^3$ would be printed + -20x^3.

Task #2: Create a Polynomial class

This class should have an ArrayList of Terms as its only instance field. There should be an accessor method for this instance field. I would suggest that you develop a method for distributing the *2x* across a polynomial. The header of that method might look like this:

```
public Polynomial distribute(Term t)
```

I would also suggest that you develop a method for subtracting 2 polynomials. The header of that method might look like this:

```
          public Polynomial subtract(Polynomial p)
```

This is by far the most challenging part of the project!

You may want to add other methods to this class if you would like it to be usable by more than just the Cebysev project.

Task #3: Create a driver called Cebysev

The main data structure in the driver class should be an ArrayList of Polynomials. The first two Polynomials of this ArrayList are given and should represent the first 2 Cebysev polynomials listed above. The code for this given information might look like this:

```
ArrayList<Polynomial> ceby = new ArrayList<Polynomial>();

ArrayList<Term> term1 = new ArrayList<Term>();
Term t1 = new Term(1,0);
term1.add(t1);
Polynomial a = new Polynomial(term1);
ceby.add(a);

ArrayList<Term> term2 = new ArrayList<Term>();
Term t2 = new Term(1,1);
term2.add(t2);
Polynomial b = new Polynomial(term2);
ceby.add(b);
```

The rest of the driver class should generate the remaining Cebysev polynomials and print all of them out.

The final output for this project should look like this:

```
 + 1x^0

 + 1x^1

 + 2x^2 + -1x^0

 + 4x^3 + -3x^1
 + 8x^4 + -8x^2 + 1x^0

 + 16x^5 + -20x^3 + 5x^1

 + 32x^6 + -48x^4 + 18x^2 + -1x^0

 + 64x^7 + -112x^5 + 56x^3 + -7x^1

 + 128x^8 + -256x^6 + 160x^4 + -32x^2 + 1x^0

 + 256x^9 + -576x^7 + 432x^5 + -120x^3 + 9x^1
```

Since Java is an object-oriented language the efficient use of the Term and Polynomial classes is mandatory, as is the recommended data types in each of the 3 classes.