CSE 41321 Homework #2
Taylor Allen
1-28-2020

a)

```
/**
* Appends a term to the end of the polynomial
* @param polynomial polynomial you want to append to
* @param coefficient coefficient you want to append to the polynomial
*/
static void appendTerm(SinglyLinkedList<Double> polynomial, Double coefficient) {
  // Append term to tail of polynomial
  polynomial.insertTail(coefficient);
}
```

b)

```
/**
* Print polynomial in polynomial format
* @param polynomial input polynomial
*/
static void display(SinglyLinkedList<Double> polynomial) {
  SinglyLinkedList<Double>.Element elem = polynomial.getHead();
  int power = polynomial.getSize() - 1;
  // Loop through each term in polynomial
  while(elem != null) {
     /**
      * All of these if/else if/else statements are just for formatting when printing,
      * for example, we print just x if the term is 1 and the power is 1 (so we don't
      * print 1.0x)
      */

     // If term is 1 and the power is not 0
     if(elem.getData() == 1 && power != 0) {
        if(power == 1) {
           System.out.print("x");
        }
        else {
           System.out.print("x^" + power);
        }
     }
     // If term is positive, element is not head, and element is not 0
```

```
        else if(elem.getData() > 0 && !(elem.getData() == polynomial.getHead().getData()) &&
elem.getData() != 0) {
            System.out.print(" + " + elem.getData());
            if(power != 0) {
                if(power == 1) {
                    System.out.print("x");
                }
                else {
                    System.out.print("x^" + power);
                }
            }
        }
        // If element is negative and element is not 0
        else if(elem.getData() < 0 && elem.getData() != 0) {
            // If element is the head of polynomial
            if(elem.getData() == polynomial.getHead().getData()) {
                System.out.print(elem.getData());
                if(power == 1) {
                    System.out.print("x");
                } else {
                    System.out.print("x^" + power);
                }
            }
            // If element is not the head of polynomial
            else {
                System.out.print(" - " + Double.toString(elem.getData()).substring(1));
                if (power != 0) {
                    if (power == 1) {
                        System.out.print("x");
                    } else {
                        System.out.print("x^" + power);
                    }
                }
            }
        }
        // Decrement power for when we go to next element in polynomial
        power--;
        // Increments to next element in polynomial
        elem = elem.getNext();
    }
}
```

c)

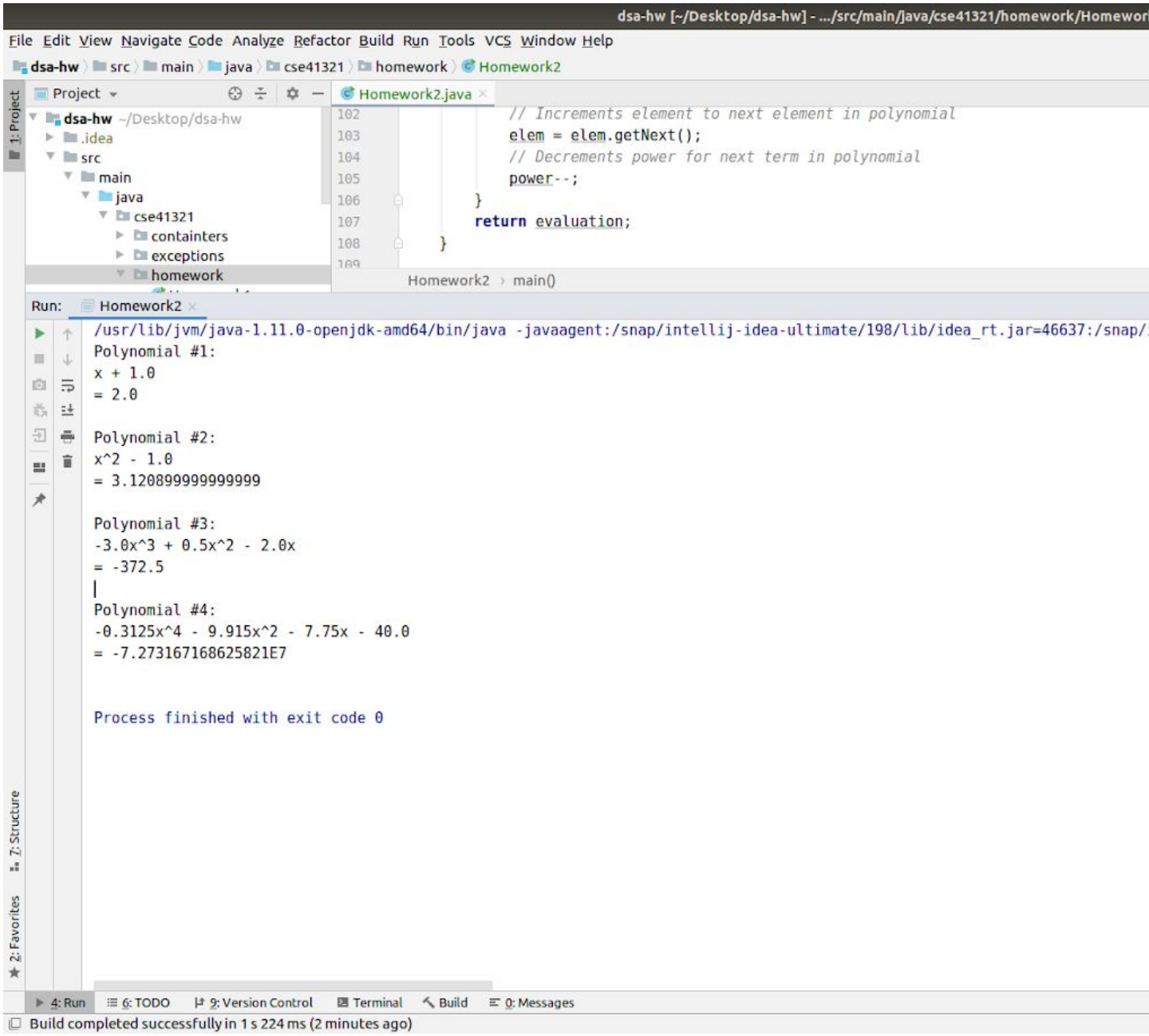
```
/**
* Evaluate the polynomial for the given value x and return the value
* @param polynomial input polynomial
* @param x input x value
* @return evaluation of polynomial
*/
static Double evaluate(SinglyLinkedList<Double> polynomial, Double x) {
  Double evaluation = 0.0;
  SinglyLinkedList<Double>.Element elem = polynomial.getHead();
  int power = polynomial.getSize() - 1;
  // Loops through each element in polynomial
  while(elem != null) {
    // Calculates value for given element, for example, 5x^2 when x is 7 is evaluated as
5(7^2)
    evaluation += elem.getData() * Math.pow(x, power);
    // Increments element to next element in polynomial
    elem = elem.getNext();
    // Decrements power for next term in polynomial
    power--;
  }
  return evaluation;
}
```

d)

```
public static void main(String[] args) {
  /**
   * Create all polynomials for testing
   * Expected polynomials:
   *
   * poly1: x + 1.0                          with x = 1.0       expected evaluation: 2.0
   * poly2: x^2 - 1.0                        with x = 2.03      expected evaluation: 3.1209
   * poly3: -3.0x^3 + 0.5x^2 - 2.0x          with x = 05.0      expected evaluation: -372.5
   * poly4: -0.3125x^4 - 9.915x^2 - 7.75x - 40.0    with x = 123.45    expected evaluation:
-72731671.69
   *
   *
   */
  SinglyLinkedList<Double> poly1 = new SinglyLinkedList<Double>();
  SinglyLinkedList<Double> poly2 = new SinglyLinkedList<Double>();
  SinglyLinkedList<Double> poly3 = new SinglyLinkedList<Double>();
```

```
    SinglyLinkedList<Double> poly4 = new SinglyLinkedList<Double>();
    SinglyLinkedList<Double> poly5 = new SinglyLinkedList<Double>();


    appendTerm(poly1, 1.0);
    appendTerm(poly1, 1.0);
    System.out.println("Polynomial #1:");
    display(poly1);
    Double poly1Evaluation = evaluate(poly1, 1.0);
    System.out.println("\n= " + poly1Evaluation + "\n");

    appendTerm(poly2,1.0);
    appendTerm(poly2,0.0);
    appendTerm(poly2,-1.0);
    System.out.println("Polynomial #2:");
    display(poly2);
    Double poly2Evaluation = evaluate(poly2, 2.03);
    System.out.println("\n= " + poly2Evaluation + "\n");

    appendTerm(poly3, -3.0);
    appendTerm(poly3, 0.5);
    appendTerm(poly3, -2.0);
    appendTerm(poly3, 0.0);
    System.out.println("Polynomial #3:");
    display(poly3);
    Double poly3Evaluation = evaluate(poly3, 05.0);
    System.out.println("\n= " + poly3Evaluation + "\n");

    appendTerm(poly4, -0.3125);
    appendTerm(poly4, 0.0);
    appendTerm(poly4, -9.915);
    appendTerm(poly4, -7.75);
    appendTerm(poly4, -40.0);
    System.out.println("Polynomial #4:");
    display(poly4);
    Double poly4Evaluation = evaluate(poly4, 123.45);
    System.out.println("\n= " + poly4Evaluation + "\n");

}
```

OUTPUT:


dsa-hw [~/Desktop/dsa-hw] - .../src/main/java/cse41321/homework/Homewor
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
dsa-hw src main java cse41321 homework Homework2
Project
dsa-hw ~/Desktop/dsa-hw
.idea
src
main
java
cse41321
containters
exceptions
homework
Homework2.java
// Increments element to next element in polynomial
elem = elem.getNext();
// Decrements power for next term in polynomial
power--;
}
return evaluation;
}
Homework2 › main()
Run: Homework2
/usr/lib/jvm/java-1.11.0-openjdk-amd64/bin/java -javaagent:/snap/intellij-idea-ultimate/198/lib/idea_rt.jar=46637:/snap/
Polynomial #1:
x + 1.0
= 2.0
Polynomial #2:
x^2 - 1.0
= 3.120899999999999
Polynomial #3:
-3.0x^3 + 0.5x^2 - 2.0x
= -372.5
Polynomial #4:
-0.3125x^4 - 9.915x^2 - 7.75x - 40.0
= -7.273167168625821E7
Process finished with exit code 0
4: Run
6: TODO
9: Version Control
Terminal
Build
0: Messages
Build completed successfully in 1 s 224 ms (2 minutes ago)