I. Testing cycle sort when it takes an array
    A. For Loops
        1. First for loop - loops through the classification yards in the railyard
            a) 0
                (1) Testing when a classification yard of 0 is inputted into the railyard
            b) 1
                (1) Testing when one classification yard of 1 is inputted into the railyard
            c) many
                (1) Testing when 3 classification yards of 2 are inputted into the railyard
            d) first
                (1) Testing that the first classification yard in the railyard works
            e) middle
                (1) Testing that the 2nd of 3 classification yards in the railyard works
            f) last
                (1) Testing that the 3rd of 3 classification yards in the railyard works
        2. Second for loop - loops through the train
            a) 0
                (1) An empty train
            b) 1
                (1) A train with one car
            c) many
                (1) A train with 8 cars
            d) first
                (1) Testing when the first train car isn't sorted correctly, but the rest are that it sorts correctly
            e) middle
                (1) Testing that when the train cars in the middle aren't properly sorted become sorted correctly
            f) last
                (1) Testing that when the train cars at the end aren't sorted correctly become correctly sorted
    B. If Statements
II. Closest sort with an array
    A. For Loops
        1. First for loop - loops through the classification yards in the railyard
            a) 0
                (1) Testing when a classification yard of 0 is inputted into the railyard

b) 1
(1) Testing when one classification yard of 1 is inputted into the railyard
c) many
(1) Testing when 3 classification yards of 2 are inputted into the railyard
d) first
(1) Testing that the first classification yard in the railyard works
e) middle
(1) Testing that the 2nd of 3 classification yards in the railyard works
f) last
(1) Testing that the 3rd of 3 classification yards in the railyard works

2. Second for loop - loops through the train
a) 0
(1) An empty train
b) 1
(1) A train with one car
c) many
(1) A train with 8 cars
d) first
(1) Testing when the first train car isn't sorted correctly, but the rest are that it sorts correctly
e) middle
(1) Testing that when the train cars in the middle aren't properly sorted become sorted correctly
f) last
(1) Testing that when the train cars at the end aren't sorted correctly become correctly sorted

B. If Statements
C.

III. Cycle Sort Linked List
A. For Loops
1. First for loop - loops through the classification yards in the railyard
a) 0
(1) Testing when a classification yard of 0 is inputted into the railyard
b) 1
(1) Testing when one classification yard of 1 is inputted into the railyard
c) many

(1) Testing when 3 classification yards of 2 are inputted into the railyard

d) first

(1) Testing that the first classification yard in the railyard works

e) middle

(1) Testing that the 2nd of 3 classification yards in the railyard works

f) last

(1) Testing that the 3rd of 3 classification yards in the railyard works

2. Second for loop - loops through the train

a) 0

(1) An empty train

b) 1

(1) A train with one car

c) many

(1) A train with 8 cars

d) first

(1) Testing when the first train car isn't sorted correctly, but the rest are that it sorts correctly

e) middle

(1) Testing that when the train cars in the middle aren't properly sorted become sorted correctly

f) last

(1) Testing that when the train cars at the end aren't sorted correctly become correctly sorted

B. If Statements

IV. Closest Sort Linked List

A. For Loops

1. First for loop - loops through the classification yards in the railyard

a) 0

(1) Testing when a classification yard of 0 is inputted into the railyard

b) 1

(1) Testing when one classification yard of 1 is inputted into the railyard

c) many

(1) Testing when 3 classification yards of 2 are inputted into the railyard

d) first

(1) Testing that the first classification yard in the railyard works

e) middle

(1) Testing that the 2nd of 3 classification yards in the railyard works

f) last

(1) Testing that the 3rd of 3 classification yards in the railyard works

2. Second for loop - loops through the train

a) 0

(1) An empty train

b) 1

(1) A train with one car

c) many

(1) A train with 8 cars

d) first

(1) Testing when the first train car isn't sorted correctly, but the rest are that it sorts correctly

e) middle

(1) Testing that when the train cars in the middle aren't properly sorted become sorted correctly

f) last

(1) Testing that when the train cars at the end aren't sorted correctly become correctly sorted

B. If Statements

V. Merge Array

A. First while loop

1. 0
   a) At a train size of 0
2. 1
   a) Train size of 1
3. many
   a) Train size of 3
4. first
   a) First traincar
5. middle
   a) 2nd traincar
6. last
   a) 3rd and final traincar

B. For loop

1. 0
   a) Compared to first element
2. 1
   a) Compared to first element
3. many
   a) Compared to many elements

4. first
    a) Compared to first element
5. middle
    a) Compared to second element
6. last
    a) Compared to third element

VI. Merge Linked List

A. First while loop

1. 0
    a) At a train size of 0
2. 1
    a) Train size of 1
3. many
    a) Train size of 3
4. first
    a) First traincar
5. middle
    a) 2nd traincar
6. last
    a) 3rd and final traincar

B. For loop

1. 0
    a) Compared to first element
2. 1
    a) Compared to first element
3. many
    a) Compared to many elements
4. first
    a) Compared to first element
5. middle
    a) Compared to second element
6. last
    a) Compared to third element

VII. Constructor

A. If Statement

1. Testing when a constructor is passing a classification yard of 0 - creates a rail yard with a classification yard of 1 because this leaves the train unchanged
2. Testing when a constructor isn't passing a single classification yard of 0 - logic proceeds normally

B. First for loop

1. 0
    a) Check that 1 classification yard is made when none are called for

2. 1
    a) Check that 1 classification yard is made when one is called for
3. Many
    a) Check that all classification yards are made when they are called for
4. First
    a) Check that the the first classification is added in a group of 3
5. Middle
    a) Check that the second classification yard is added in a group of 3
6. Last
    a) Check the 3rd classification yard is added in a group of 3

C. Second for loop
1. 0
    a) If no tracks are called for one is made as default
2. 1
    a) If one track is called for one is made
3. Many
    a) As many tracks that are called for are made
4. First
    a) The first track is made when three are requested
5. Middle
    a) The second track is made when three are requested
6. Last
    a) The third track is made when three are requested

VIII. Main

A. To make sure that the main method worked without using junit testing I simply typed “java RailYard cycle 3 5 2 4 banana apple cherry pear orange pineapple blueberry peach cantalope apricot durian honeydew cranberry grapefruit watermelon dragonfruit kiwi grape quince raspberry kumquat plum” into the interactions pane and compared what I got by hand with what the computer printed on the console and they were the same. I also made sure that the try catch worked by typing gibberish onto the screen in various different permutations and every time it printed out my catch statement telling the user to reconsider what they’ve typed.

IX. Works for Closest but not for Cycle

A. “elephant”, “cat”, “indigo”, “dog”, “llama” can be sorted correctly by closest and not cycle when the rail yard consists of one classification yard containing 2 tracks. Closest sorts it into “cat”, “dog”, “elephant”, “indigo”, “llama” but cycle sorts it incorrectly into “cat”, “elephant”, “dog”, “indigo”, “llama”. Cycle doesn’t work as well sometimes because as opposed to closest it doesn’t always put the objects nearest to the ones it should.

X. Works for Cycle but not for Closest

A. 5, 2, 7, 3, 2, 5, 9, 9 can be sorted correctly by cycle and not closest when the rail yard consists of two classification yards the first containing 2 tracks and the second containing 3 tracks. Cycle sort sorts it into 2, 2, 3, 5, 5, 7, 9, 9 but closest sorts it incorrectly into 2, 3, 5, 2, 5, 7, 9, 9. Closest doesn't work as well sometimes because when it is sending a number to an empty row the way it is programmed is to send it to the last empty row.