<Mangesh Raut>

<May 18th 2022>

Week8Meet - 10 pts

Turn in on BBL as soon as complete, but before end of day Friday following the lecture.

Answer these questions as we progress through the meeting.

1. What goes into a tester for Events? Who was in your room?

In Event class the methods like getters setters in short, the methods of event class goes into the tester to check it is working properly or not. Event class has connection to other classes like Ride, Watch, and Shop these classes also has classes so all the classes connected to event class will go for the tester to test the working flow.

Mangesh, Akshay, Dayo, Ron

1. Organize your thoughts into a high-level list of tasks the program should perform:

Event class is the main class so it should work perfectly then other classes also works fine. Event class has connection to every classes. Gift, shop, and exit the classes where they have subclasses, and their connections also works fine. If they found any exceptions like array or string file exceptions, we have to use try catch block to caught them and works properly.

Family – Agenda – Event Ride – Watch – Shop – Food – Exit these is my way of seeing flow.

1. What is the purpose of the try/catch/finally structure?

Try and catch is used for exceptions. In try block whatever exceptions are there it will exist that will catch using the catch. Catch block used after the try block.

The try-catch statement consists of a try block followed by one or more catch clauses, which specify handlers for different exceptions. When an exception is thrown, the common language runtime (CLR) looks for the catch statement that handles this exception.

The finally block always executes when the try block exits. This ensures that the finally block is executed even if an unexpected exception occurs.

1. Selection sort the following (show the state of the list on each iteration):  
   5 9 7 3 6 1 4 2 8

[ ] [5, 9, 7, 3, 6, 1, 4, 2, 8 ] (8)

[ 1 ] [ 5, 9, 7, 3, 6, 4, 2, 8] (7)

[ 1, 2 ] [ 5, 9, 7, 3, 6, 4, 8] (6)

[ 1, 2, 3] [ 5, 9, 7, 6, 4, 8] (5)

[ 1, 2, 3, 4] [ 5, 9, 7, 6, 8] (4)

[ 1, 2, 3, 4, 5] [ 9, 7, 6, 8] (3)

[ 1, 2, 3, 4, 5, 6] [ 9, 7, 8] (2)

[ 1, 2, 3, 4, 5, 6, 7] [ 9, 8] (1)

[ 1, 2, 3, 4, 5, 6, 7, 8] [ 9] (0)

[ 1, 2, 3, 4, 5, 6, 7, 8, 9]

1. Insertion sorts the following (show the state of the list on each iteration):  
   5 9 7 3 6 1 4 2 8

[5, 9, 7, 3, 6, 1, 4, 2, 8] [ 1, 5, 9, 7, 3, 6, 4, 2, 8] [ 1, 2, 5, 9, 7, 3, 6, 4, 8]

[5, 9, 7, 3, 1, 6, 4, 2, 8] [ 1, 5, 9, 7, 3, 6, 2, 4, 8] [ 1, 2, 5, 9, 3, 7, 6, 4, 8]

[5, 9, 7, 1, 3, 6, 4, 2, 8] [ 1, 5, 9, 7, 3, 2, 6, 4, 8] [ 1, 2, 5, 3, 9, 7, 6, 4, 8]

[5, 9, 1, 7, 3, 6, 4, 2, 8] [ 1, 5, 9, 7, 2, 3, 6, 4, 8] [ 1, 2, 3, 5, 9, 7, 6, 4, 8]

[5, 1, 9, 7, 3, 6, 4, 2, 8] [ 1, 5, 9, 2, 7, 3, 6, 4, 8]

[1, 5, 9, 7, 3, 6, 4, 2, 8] [ 1, 5, 2, 9, 7, 3, 6, 4, 8]

[ 1, 2, 5, 9, 7, 3, 6, 4, 8]

[ 1, 2, 3, 5, 9, 7, 6, 4, 8] [ 1, 2, 3, 4, 5, 9, 7, 6, 8] [ 1, 2, 3, 4, 5, 9, 7, 6, 8]

[ 1, 2, 3, 5, 9, 7, 4, 6, 8] [ 1, 2, 3, 4, 5, 9, 7, 6, 8] [ 1, 2, 3, 4, 5, 9, 6, 7, 8]

[ 1, 2, 3, 5, 9, 4, 7, 6, 8] [ 1, 2, 3, 4, 5, 6, 9, 7, 8]

[ 1, 2, 3, 5, 4, 9, 7, 6, 8]

[ 1, 2, 3, 4, 5, 9, 7, 6, 8]

[ 1, 2, 3, 4, 5, 6, 9, 7, 8] [ 1, 2, 3, 4, 5, 6, 7, 9, 8] [ 1, 2, 3, 4, 5, 6, 7, 8, 9]

[ 1, 2, 3, 4, 5, 6, 7, 9, 8] [ 1, 2, 3, 4, 5, 6, 7, 8, 9]

1. In a list of 100 items, what is the WORST case for linear search? The BEST case? The AVERAGE case?

Worst case O(n) which is O(100)

Best case O(1)

Average case O(n/2) which is O(100/2) = O(50)

1. In a list of 100 items, what is the WORST case for binary search? The BEST case? The AVERAGE case?

Worst case O(log n) which is O(log 100)

Best case O(1)

Average case O(n/2) which is O(log 100)

1. What is your current understanding of ‘Order of n’ O(n)?

Big O notation is a mathematical notation that describes the limiting behavior of a function when the argument tends towards a particular value or infinity.

Big O is the complexity of the method which shows the runtime of the method. When we increase the size or number it will also increase the runtime time. N is the number of that shows the how many numbers it will take and that is the runtime of that method. Example if sort 8 number its O(8).

Reflect on your learning and your needs. After this class meeting, what topics do you feel like you learned and what topics do you feel like you need more information on to learn?

I learn about the complexity of linear and binary search they have worst average and best cases. In sorting I see the differences in insertion and selection sort how they perform and what is coding part. Also, the try catch and finally block and there use. I would like to learn more about the try catch.