Sorting

 $\bullet \bullet \bullet$

Cis 110 Recitation 7/26/17

Sorting recap

- Binary search: Divide the set of things you're searching in half at each step

 Only works if the elements are sorted
- Insertion sort: take each element in the unsorted part of the array and *insert* it into the sorted part in the correct place
- Selection sort: *Select* the smallest element from the unsorted part and put it in the next space to be sorted (swapping the two elements)

Objects recap

- static vs. non-static
- public vs. private
- final
- Calling static functions
- Calling instance methods

Exercise: sort an array of Cars by speed

- Car class has the following API (application program interface):
 - Car(String make, String model)
 - Car(String make, String model, int speed)
 - o void start()
 - void stop()
 - void accelerate(int x)
 - o boolean isMoving()
 - String toString()
 - o int getSpeed()
 - o static int getNumCars()
- Write a function, sortCarsInsertion(Car[] cars), that sorts an array of Cars in ascending order by speed using insertion sort
- Write a function, sortCarsSelection (Car[] cars), that does the same thing with selection sort

Bonus objects exercise

Write a Car method, public void crash (Car other), which simulates the two cars colliding

- Find the average speed of the two cars
- Set each car's speed to that average
- Example: Car A is traveling at 30mph, Car B is going 60mph
 - \circ When they crash, some of Car B's kinetic energy is transferred to Car A
 - Both are now traveling at 45mph ((30 + 60) / 2)