Project: Let's Sort It Out

For this project, you will create a class that implements comparable and then sort a list of them using various sorting algorithms.

What are the requirements for the project?

The following are **required** to earn points for the project:

Other than your **Main** file, you should create 1 class called **Course**. Do not use other names.

Course Class (25%)

Create a class called **Course** which stores a department, course number and course name. For instance "CS", 241, "Data Structures and Algorithms".

- This class must implement **comparable** to compare it to another Course so they can be sorted: CS 121 would go before CS 241, however CIS 151 would go before CS 121 because CIS is earlier in the alphabet than CS. MAT 100 would come after CS 241 because MAT is later in the alphabet than CS.
- This class should also include a **toString** method which will return a String in this example format: CS 241: Data Structures and Algorithms.

Everything below should be inside **Main.java**

Print (5%)

Write a method called print that takes an ArrayList<Course> and prints the list. This will come in handy in your testing. You can test if your comparable works by using Collections.sort and then printing the result.

```
public static void print(ArrayList<Course> a)
```

Bubble Sort (20%)

Write a method called bubbleSort that takes an ArrayList<Course> and performs the bubble sort. You may refer to the book, your notes or this Wikipedia article: https://en.wikipedia.org/wiki/Bubble_sort

```
public static void bubbleSort(ArrayList<Course> a)
```

Insertion Sort (20%)

Write a method called insertionSort that takes an ArrayList<Course> and performs the insertion sort. You may refer to the book, your notes or this Wikipedia article: https://en.wikipedia.org/wiki/Insertion_sort

```
public static void insertionSort(ArrayList<Course> a)
```

Selection Sort (20%)

Write a method called selectionSort that takes an ArrayList<Course> and performs the selection sort. You may refer to the book, your notes or this Wikipedia article: https://en.wikipedia.org/wiki/Selection_sort

```
public static void selectionSort(ArrayList<Course> a)
```

Report (10%)

In the comments at the top (see Main.java below), answer separately for each (Bubble Sort, Insertion Sort, Selection Sort). **Based on your code**, What is the worst case big O and what kind of input would cause that. What is the best case big O and what kind of input would cause that? Be sure to write **in your own words**!

How do I submit my work?

Your project must compile! Non-compiling projects will get a 0. When you are done, you must upload your code to the **Assignments** area. This must be code files. Do not take pictures of code or email code.

Projects received 1 minute late are considered late. Start uploading your project at least an hour before the deadline to avoid a point deduction. If there are any issues with uploading your project, you must **email me before the due date**. Email cguida@pace.edu from your @pace.edu email address.

While I check email regularly, **do not expect a response over the weekend or close to deadlines**.
Late projects will have **10 points deducted per day**. Late projects will **not be accepted after 2 days**.

Plagiarism, cheating and other ways you will get a 0 on this project:

Your code must be your own code. Do not use Chegg, CourseHero, ChatGPT or any other websites like these. If you watch a YouTube video or other online resource and put in their code, that's not your code. **That is someone else's code**. You will get a 0. Do not share your code or collaborate. **Both** of you will get a 0.

Help, I'm stuck!

Start the projects early, if you are stuck, **reach out to me** cguida@pace.edu, stop by office hours and make use of the **Seidenberg** tutors for help.

Starter Code: Main.java

```
/*
  Your Name:  [YOUR FULL NAME]
  Pace Email: [YOUR PACE EMAIL ADDRESS]

  Based on YOUR code.

  Bubble Sort:
  What is the worst case big O and what kind of input would cause that?

  What is the best case big O and what kind of input would cause that?

  Insertion Sort:
  What is the worst case big O and what kind of input would cause that?

  What is the best case big O and what kind of input would cause that?

  Selection Sort:
  What is the worst case big O and what kind of input would cause that?

  What is the best case big O and what kind of input would cause that?

*/

class Main {
  public static void main(String[] args) {

  }
}
```