Homework 1, CMSC 3103 OOP, Fall 2012

Due: September 22, **Saturday**, **11:59PM** (No late submission accepted)
Max points allocated: 50 points

Write a Java program similar to a hangman game given as a demo.

- Login `cs.uco.edu`: username: `oop`, password: `fall2012`
- Run the demo as follows:

```
    java -jar hangman.jar
```

Program requirements:

- Study Java APIs: `ArrayList`, `String`, `StringBuffer`
- Define at least 20 words as `String`, and use them as the list of words for the game key
- Store the above 20 `Strings` in `ArrayList` and choose one randomly as game key.
- You may define multiple classes for this assignment, but the main Class (the class with 'main' method') must be named as `Homework1`
- Be sure to follow Java naming conventions and use proper indentations.
- **Display the game key** for you to debug and for me to evaluate your program.
- Students must use the OO design paradigm; use of strict queries/commands (minimal requirement)

Turn-in:

- Place all source files in a folder and compress the folder.
- Submit the compressed folder via D2L.

Common mistakes that result in deductions

- The main class name is not Homework1.
- The game key is not displayed.
- An array is used instead of **ArrayList**.
- Java naming convention is violated.
- Used/Unused letters are not correctly displayed.
- The user interface is different from that of demo (e.g., the program quits after a certain number of trials, etc)

Grading:

- Partial credits will not be given if the program cannot be played. In other words, the program must run correctly for the core part of the hangman game.