Half Exam 1 Name:

This is a 25-minute exam. The exam is printed double sided. If you use a note-sheet (prepared by yourself), please turn it in with your exam.

1. (10 point) ***Describe*** the difference between Coupling and Cohesion. ***Make clear*** which is which, and ***make clear*** what "higher" means for each.

2. (20 points) ***Re-write*** the code
   ```
   javax.swing.Timer timer = new javax.swing.Timer(1000,
   label.setText(“”+new Date() ) );
   ```
   to use the full anonymous inner class syntax instead of a lambda expression. Recall that a timer takes an EventListener as an argument, which has a single method `void actionPerformed(ActionEvent e);`

3. (20 points) ***Draw*** a UML diagram for a weather app that has multiple "views" of the weather that update each time the weather changes. ***Use*** the Observer pattern. ***Include*** all classes, class names, and relationships. ***Annotate*** interfaces and abstract classes. You do not need to include method names.

4. (10 points) ***Describe*** how **programming to an interface** makes code more extensible. Use the definition of **programming to an interface** from class.

5. (40 points) The following method is part of a web-page search tool. It uses an insertion sort to place each web-page in the right spot in the list.
    a. ***Edit*** the following code to extend the method to ***use*** the strategy pattern to allow pages to be ranked in different ways (see part b). `pages` is an instance variable holding all the pages that this engine can search.

```
public List<Page> search(String query                                    ) {
    List<Page> results = new ArrayList<>();
    for(int oldIndex = 0; oldIndex < pages.size(); oldIndex++) {
        int resIndex = 0;
        Page page = pages.get(oldIndex);
        while (resIndex < results.size()
                && page.numMatchingWords(query) <=
                        results.get(resIndex).numMatchingWords(query)) {
            newIndex++;
        }
        results.add(newIndex, page);
    }
    return results;
}
```

b. ***Write*** the interface used by your code above.