Name: ______________________

Rubric for Assignment #4

Criteria		Unsatisfactory	Acceptable	Good	Exceptional	Marks
		0	1	2	3	
Functionality	**Employee Data File**	- file contains too little data or data is non-existent - not all fields exist in a record	- the data file contains less than 10 records - no field contains spaces	- the data file contains at least 15 records - some fields contain spaces - records are already sorted by some fields - no duplicate records or fields exist	- the data file consists of at least 20 records with a large range of different values and casing - many fields contain whitespace - all records are sufficiently randomized for sorting by different fields - duplicate records and/or fields exist	*x* **2** ______
	File Sorting	- employee file cannot be sorted by any field - many large errors exist	- the file can be sorted by one field - many small or a few large errors exist in sort	- the file can be sorted by almost any field - a few small errors exist in sort	- the file can be sorted by any field - no errors exist	*x* **4** ______
	Merge Sort	- does not use Merge Sort	- uses only in memory version of Merge Sort - some errors exist	- uses Merge Sort with temporary files - still uses arrays or too much memory - a few errors exist	- uses Merge Sort with temporary files and uses a minimal amount of memory - no errors exist	*x* **4** ______
					Sub-Total	______ 30

Note: 30 x 30% = 9

Note: The following aspects of the program will only be graded if you receive more than 30% on the functionality rubric.

Source Code	**Readability**	- source code is poorly organized and very difficult to read	- source code can be read, but is hard to follow	- source code is fairly easy to read, but is hard to follow in some areas	- source code is exceptionally well organized and easy to follow	________
	Reusability	- source code cannot be reused - no functions or classes used	- small sections of code could be reused	- large portions of code could be reused with some modifications	- source code could be easily reused with little modifications	________
	Efficiency	- contains large portions that could have been easily reduced using a different method - a lot of code is duplicated, copy/pasted	- tried some methods to improve efficiency - can explain what they attempted	- employed good ideas to improve efficiency - can point out where other improvements could be made	- very clean and efficient code - can propose new ideas for improvement	________
	Comments	- little to no comments used	- comments are used, some are meaningful and easily understood - some files and functions have headers	- comments are used extensively, most are meaningful and easily understood - most files/functions have headers	- not over/under commented - comments are meaningful and easily understood - files/functions have headers - is self-documenting	________
	Naming Convention	- no standard naming convention followed	- a standard naming convention was used for part of the program, but deviated often	- a standard naming convention was used for most of the program and deviated very little	- industry standard naming convention used throughout the program	________
	Consistency	- no consistency in formatting or layout of source code	- source code formatting was fairly consistent, but contained some inconsistency with whitespace, brackets, etc	- source code formatting was very consistent with respect to whitespace, brace brackets, parentheses, etc	- source code formatting never deviated from the programmer's layout	________

Sub-Total	________ **18**
Total	________ **48**

Assignment: ______________________________