F21AS Advanced Software Engineering Coursework 2013-14 – Edinburgh Campus

Stage 1 Assessment	**Mark out of 70**	**57**
	Mark out of 50	40.7
GROUP	sd196	Dalviken, Stian
	kk249	Khamrun, Kamontorn
	mpk31	Kopacz, Marcin

OVERALL	A very good submission, meeting all the criteria. Some comments for improvement.

TOPIC	**MAX**	**MARK**	**COMMENTS**
Accurate processsing, output to text file	10	8	Nice output file though try to line up decimal points and always show 2dp for currency. Menu nicely presented but usually start with the starter and end with the drinks. Own stats are min/max/ave/total. User input only has one chance - better to allow multiple bill printing? Error message for nonnumeric and no-table entries.
Good OOP design, clear modular well-commented code, clear class diagrams	10	9	Good OOP design, clear class diagram, methods have comments. Would be nice also to include a little guidance in comment form through reporting methods which are complex.
Use of data structures	10	8	A selection of data structures used, mostly for appropriate reasons (not sure about HashSet of orders). Good to find the Collections.frequency method, though some ordering of the dishes would have been nice. I was hoping groups would use set differences for the Not Ordered - good! Table summaries use TreeSet to order menu items, then looping.
Testing of input files and use of exceptions (including in constructor)	10	9	Good use of exceptions and well explained in the report. Program does not cater for orders not on the menu, but negative prices are rejected.
JUnit tests	10	8	A good set of tests, but when testing a calculation, just test that all the paths are correct, rather than testing for incorrectness. Why not check for invalid quantity when creating an order rather than when adding to a collection? Includes a test for an exception.
Discussion on development plan	20	15	Reasonably detailed plan, with discussion in report
TOTAL	70	57	