

Basic Preparation	2	0-1	Ready to go at the start of section
Cleverness	2	0-2	The hardest points on the rubric
Code Submission	4	0-2	Submitted correct content on time and to the correct location in the repository
Decomposition	4	0-2	Project is adequately decomposed to different classes and methods
Documentation	4	0-2	Comments for each class and each function are clear and are following style guides
Effort	2	0-2	Perform considerable amount of work
Naming	2	0-2	Variable names and method names are readable and are following Java conventions
Overall Design	4	0-2	Have nice approaches and structures in overall
Participation	5	0-2.5	Interact with the group 2 times (ask a question, make a comment, help answer a question, etc.)
Presentation	4	0-2	Present the code clearly
Requirements - Refine UI	5	0-2.5	<ul style="list-style-type: none"> • 2 points - look clear and presentable • 2.5 points - look nice, beautiful and adaptive
Requirements - Solving linear least square	5	0-2.5	<ul style="list-style-type: none"> • 2 points - basic linear least square • 2.5 points - support refinement by removing one item with largest error
Requirements - Polynomial interpolation	5	0-2.5	<ul style="list-style-type: none"> • 2 points - exact fitting • 2.5 points - best fitting when degree specified
Requirements - Vector	5	0-2.5	<ul style="list-style-type: none"> • 2 points - vector arithmetic (linear, inner product) • 2.5 points - cross product

Testing	5	0-2.5	Unit tests written for all new code & expanded last week's tests if necessary <ul style="list-style-type: none"> • 2 points - 80% of the new code is tested and the tests cover edge cases for each function. • 2.5 points - 95% of the new code has been tested and every possible edge cases has been tested for.
Schedule	2	0-1	<ul style="list-style-type: none"> • Revised final project schedule if necessary
Total	63		