Appendix A Grading Rubric for Track A

When there is one insufficient present in the final grading rubric, the student fails this assignment. **Assessor:** **Total score:**

Be very careful, if you only score grading item with only 'sufficient' and without the bonus, your total grade is still not enough for you to pass(4.6/10). Total score is capped at 10.

Sections	Items	Grading standard				Score
Report (4)	Page limit (0.25)	**Insufficient (0)** Main body less than 3 pages or more than 25 pages	**Sufficient(0.1)** Main body slightly more than 21 pages	**Good(0.25)** Main body within the page limit		
	Structure (0.35)	**Insufficient (0)** Missing major part(s) of the report with no justification. Chaotic organization of the report.	**Sufficient(0.2)** Most components of a professional report are there. Student can justify the missing part. Not-so-logical content organization.	**Good(0.35)** The report is structured in a professional way with all necessary components. Logical organization of the contents.		
	Readability and Quality (0.4)	**Insufficient (0)** Overly compressed formatting. Hard to read, bad English sentences. Lots of grammar errors. Vague pictures. No title, numbering, explanation for the figures and tables.	**Sufficient(0.3)** Clear, readable, report formatting. Readable pictures, diagrams, tables, etc. with title. Some grammar errors occur.	**Good(0.4)** Clear, easy-to-read, professional report formatting. High quality pictures, figures, diagrams, tables, etc. with meaningful titles, proper legend and necessary information like axis names and units. Very few (<4) grammar errors.		
	Necessary points to explain (2.5)	**Insufficient(0)** Missing more than 5/11 necessary elements mentioned. Little or no explanation and justification provided. Poor logic and explanation. Lots of conceptual mistakes. The report is constantly missing the points and self-contrasting results.	**Sufficient(1.15)** 5 to 2 necessary elements missing. Few explanations and justification provided. Or the explanation does not stand logically or very difficult to understand. The report is occasionally missing the point. Explanations and justifications are sometimes not backed-up with facts and source. Little conceptual mistakes were made.	**Good(1.85)** Only 1 or no necessary element missing. The explanations and justifications are often built upon sound results and proofs. The report is mostly on point and convincing. Occasional conceptual mistakes were made.	**Excellent(2.5)** no necessary element missing. The explanations and justifications are logical and always built upon sound results and proofs. The report is always on point and convincing. No conceptual mistakes made.	

	References (0.5)	**Insufficient (0)** No or very little reference given. Directly copying long text from referenced material.	**Sufficient(0.3)** Proper and correct references are given. Some referencing style is inconsistent. Some source is doubtful or not trustworthy.	**Good(0.5)** Correct, proper, trustworthy reference with consistent style.	
Code (4)	Style & readability (0.75)	**Insufficient (0)** In-consistent style, very difficult to read. No or almost no comment provided.	**Sufficient(0.6)** Mostly consistent, clear looking code. Occasional comments provided for better understanding the code. Or a bit too much comments flooding in the script. Could be too many or too few lines of code.	**Good(0.75)** Consistent coding style. Clear-looking script. Not too many lines of code. Essential and helpful comments included in the script with suitable amount.	
	Quality (0.75)	**Insufficient (0)** Critical bugs, or un-usable, or completely un-readable, or completely un-testable code. Most identifiers make no sense. Your script took too long to run. Your script cannot accomplish either of the actions: encrypt, decrypt, or hack the cipher and obtain the key.	**Sufficient(0.75)** Identifiers make sense. Well informed input output of the self-defined methods and classes. Your script is executed within the required time. Your script can accomplish all of the following actions: encrypt, decrypt, and hack the cipher and obtain the key.		
	Test cases (1.5)	**Insufficient (0)** Passes less than 7 out of 10	**Sufficient(0.65)** Passes 7 to 9 out of 10.	**Good(1.5)** Passes 10 out of 10.	
	Documentation (0.5)	**Insufficient (0)** No documentation present. Documentation make no sense at all. Critical mistakes in documentation.	**Sufficient(0.35)** Readable, make-sense documentation.	**Good(0.5)** To the point documentation. Clearly explains every thing needed for using the software.	
	Security (0.25)	**Insufficient (0)** There is a serious security flaw in the code, making it a critical threat to run in certain environment. If the security flaw is inevitable, it's not properly documented or mentioned in the report or the readme document.	**Sufficient(0.25)** There's no obvious security threat caused by the code itself. It's safe to run in any computer. If the security flaw is inevitable, it's properly documented or mentioned in the report or the readme document.		
	Able to run? (0.25)	**Insufficient (0)** The code always impose errors during runtime.	**Sufficient(0.25)** The code is able to execute with no error.		
Defence (1)	Report (0.5)	**Insufficient(0)** The report, knowledge, and research insights appear strange to the student. Mostly off the point.	**Sufficient(0.15)** The student can roughly explain the report and relevant research process and knowledge. Mostly to the point.	**Good(0.5)** The student can precisely explain the report and relevant research process and knowledge. Fluent and to the point.	

	Code (0.5)	**Insufficient(0)** The code and knowledge appear strange to the student. Mostly off the point.	**Sufficient(0.15)** The student can roughly explain the code and relevant knowledge. Mostly to the point.	**Good(0.5)** The student can precisely explain the code and relevant knowledge. Always to the point.	

BONUS (2)	There is a logical research process on the bonus problem (0.25)	**No(0)**	**Yes(0.25)**	
	Students can demonstrate their knowledge acquired on the bonus topic (0.25)	**No(0)**	**Yes(0.25)**	
	A tool conducting enc-dec operations is programmed (0.25)	**No(0)**	**Yes(0.25)**	
	The tool can work properly and the student can demonstrate that (0.5)	**No(0)**	**Yes(0.5)**	
	The student can find a way to crack the keys of generic substitution cipher and demonstrate this with a self made python tool (0.75)	**No(0)**	**Yes(0.75)**	