	Criteria	Sub-criteria	1	2	3	4-5	6
1	**Ability to demonstrate design solution**	Construct correct pseudocode	Unable to construct	Able to construct but with wrong logic	Able to construct correctly but with missing or unnecessary elements	Able to construct correctly and use proper elements	Able to construct correctly, use proper elements, documentation and structure
2	**Ability to analyse problem and identify requirements**	Identify correct input / output	Unable to identify any input and output	Able to identify only one input or output	Able to identify correctly some input and output	Able to identify correctly all input and output	Able to identify correctly all input and output and manage them optimally
3	**Ability to apply required data type or data structure**	Appropriate choice of variable names or data structure	Unable to identify required data type or data structure	Able to identify required data type or data structure but does not apply correctly	Able to apply required data type or data structure but does not produce correct results	Able to apply required data type or data structure and produce partially correct / partial results	Able to apply required data type or data structure and produce correct results
4	**Ability to apply required flow control structure**	Correct choice of sequential, selection or repetition control flow structure	Unable to identify required control structure	Able to identify required control but does not apply correctly	Able to apply required control structure but does not produce correct results	Able to apply required control structure and produce partially correct/partial results	Able to apply required control structure and produce correct results
5	**Ability to optimise**	Correct choice of optimal elements	Use several non optimal elements and result is wrong	Use a few non optimal elements and result is wrong	Use several optimal elements and result is correct	Use mostly optimal elements	Program is fully optimised
6	**Ability to modularise**	Construct modularised programs	Unable to modularise	Able to use one or a few modularised elements, but not to write a modularised program	Able to write modularised programs, but with problems in the structure and/or functioning	Able to write modularised programs	Able to write fully modularised programs. Modularisation is well managed/structured
7	**Ability to run/debug**	Free from syntax, logic, and runtime errors	Unable to run program	Able to run program but have several logic errors	Able to run program but have one or a few logic errors	Able to run program correctly without any logic error and display partially appropriate output	Able to run program correctly without any logic error and display appropriate output
8	**Ability to produce readable program**	Comment / Description	No documentation	Documentation is simple comment in code	Documentation is simple comments embedded in code and header describing input and output	Documentation is detailed comments and/or header that is useful in understanding the code. Modularised elements display documentation	Documentation is well- written and clearly explains what each part/element of the code is accomplishing. The header contains info about the developer, the code release and updates, licence, etc.
9		Indentation / Naming Convention	Unable to organise the code	The code is poorly organised and very difficult to read	The code is readable only by a person who already knows its purpose	The code is fairly easy to read	The code is extremely well organized and easy to follow
10	**Ability to complete exercise**	Program is complete (input/action(s)/output)	Unable to collect the input	The input is collected but it is not used	The input is collected and used, but output is not generated	The input is collected and used, but the program does not display correct results	The input is collected and used, and correct output is generated