**Section: Programming (Advocate: Manish Gadhvi)**

**Provide a definition of what an algorithm is and outline the process in building an application.**

|  |
| --- |
| <https://github.com/LBruni98/Glossary-of-Tech-Terms/blob/master/README.md#1-algorithm>  <https://github.com/LBruni98/Glossary-of-Tech-Terms/blob/master/README.md#process-of-implementation> |
| The two links contain the criteria for the task at hand. The top link is a glossary of tech terms that contains a full process of building an application, written after creating a certain program. The second link is a glossary of tech terms, that contains the definition of an algorithm. It is located towards the bottom of the page after the project management definitions. |

**Give explanations of what procedural, object orientated and event driven paradigms are; their characteristics and the relationship between them.**

|  |
| --- |
| <https://github.com/LBruni98/Glossary-of-Tech-Terms/blob/master/README.md#2-paradigms> |
| The link above is the glossary of tech terms, which contain explanations of each individual paradigm as well as each of their characteristics and relationships between them. The definitions are located at the bottom of the document, with explanations of each starting with Procedural paradigms. |

**Write a program that implements an algorithm using an IDE.**

|  |
| --- |
| <https://github.com/LBruni98/High-Low-Card-Game/blob/master/main.cpp>  <https://github.com/LBruni98/High-Low-Card-Game#flowchart>  <https://github.com/LBruni98/High-Low-Card-Game#ide> |
| The links above lead to separate parts of the second project repo. Each part of the repo contains subsequent parts of the criteria. The first link is the code of the code of the program, where the algorithm is written out and can be run in any IDE that supports C++. The second link is the flowchart that outlines the algorithm that will be used within the project, detailing the functions and features that will be included in the program. The third link is a screenshot that shows the IDE in use with the algorithm being written inside it. |

**Explain the debugging process and explain the debugging facilities available in the IDE.**

|  |
| --- |
| <https://github.com/LBruni98/Glossary-of-Tech-Terms/blob/master/README.md#debugging-process-and-debugging-facilities> |
| The link above, in my opinion, fulfils the criteria above. The link is a glossary of tech terms and this section explains the debugging process with how it helps the program. There is also the common debugging facilities that are included in the IDE, with explanations on what they do and how they help. |

**Outline the coding standard you have used in your code.**

|  |
| --- |
| <https://github.com/LBruni98/High-Low-Card-Game#coding-standards> |
| The link describes a coding standard for a specific language, in this case it’s C++, and it follows up on how important it is to use a standard. Along with that, it explains how I used it to reflect on my code. The additional link shows the .cpp file showcasing the standard I followed. |

**Determine the steps taken from writing code to execution.**

|  |
| --- |
| <https://github.com/LBruni98/Glossary-of-Tech-Terms#ide> |
| Within the wiki, the process of creating the project is described. The process showcases very step I took to create the program. I find that this is suitable evidence as it also states how I wrote the code of the program to how I was able to the initial execution of the program. |

**Analyse the common features that a developer has access to in an IDE.**

|  |
| --- |
| <https://github.com/LBruni98/Glossary-of-Tech-Terms/blob/master/README.md#analysis> |
| The links contain the projects that describes the IDEs used and the features that they have. I find that this is suitable evidence, because within the documentation of each project, each IDE has been described, including its features. Trace ball describes the features that notepad can do, though not a proper IDE and the High-low project and Anagram solver both have descriptions of its IDE, repl.it. |

**Use the IDE to manage the development process of the program.**

|  |
| --- |
| <https://github.com/LBruni98/High-Low-Card-Game#ide> |
| Here, the IDE is used to help create the program. It showcases how the code was written as well as the functions and algorithms being created and used. The IDE allowed for features to help the development and debugging of the program, to which screenshots of the IDE are located throughout the documentation. |

**Evaluate how the debugging process can be used to help develop more secure, robust applications.**

|  |
| --- |
| <https://github.com/LBruni98/Glossary-of-Tech-Terms/blob/master/README.md#debugging-process-and-debugging-facilities> |
| The glossary has an evaluation part as well as an overall description of debugging. What it reads is how the debugging process can help make secure and more functional applications. The evaluation also showcases other benefits that comes with debugging as well as my experience and how it helped me with development. |

**Examine the implementation of an algorithm in a suitable language. Evaluate the relationship between the written algorithm and the code variant.**

|  |
| --- |
| <https://github.com/LBruni98/Anagram-Solver/blob/master/README.md#development-of-code> |
| The project showcases the project algorithms being implemented. The project’s documentation describes the implementation of the program within C++ and what each algorithm is capable of doing. Also there is a flowchart detailing the algorithm and how it differentiates over the implemented variant. |

**Critically evaluate the source code of an application which implements the programming paradigms, in terms of the code structure and characteristics.**

|  |
| --- |
| <https://github.com/LBruni98/Anagram-Solver/blob/master/README.md#evaluation> |
| The project shows an evaluation of the source code over different forms, such as coding standards and the evaluation of the code against the flowchart. It also goes into detail on the coding structure and the reasons for the code layout. |

**Evaluate the use of an IDE for development of applications contrasted with not using an IDE.**

|  |
| --- |
| <https://github.com/LBruni98/Glossary-of-Tech-Terms/blob/master/README.md#comparison-of-ides-and-non-ide-software> |
| The link above is to a separate heading of the glossary of tech terms. This section goes into the comparison of IDEs and non-ide software. The comparison goes in depth into its features, the ease of development and some other topics. |

**Critically evaluate why a coding standard is necessary in a team as well as for the individual.**

|  |
| --- |
| <https://github.com/LBruni98/Glossary-of-Tech-Terms/blob/master/README.md#coding-standards> |
| The link above leads to the glossary of tech terms, under the coding standards heading. The link provided shows the project documentation, which within showcases an of the how the coding standard helps the team with readability and editing. Also described is how it helps the individual, being able to understand the code and write it effectively. |