**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/Trace-Ball/wiki/Project-Wiki---Trace-Ball#process-of-implementation>  <https://github.com/LBruni98/Glossary-of-Tech-Terms#programming-terms> |
| The two links contain the criteria for the task at hand. The top link is a wiki of the project that contains a full process of building an application, written after creating the trace ball 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#programming-terms> |
| 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/Trace-Ball#retrieving-mouse-co-ordinates>  <https://github.com/LBruni98/High-Low-Card-Game#randomly-dealt-cards>  <https://github.com/LBruni98/High-Low-Card-Game#betting>  <https://github.com/LBruni98/Anagram-Solver#reading-the-file-and-word-comparison> |
| The links above are three projects that showcase their algorithms. Each of which were implemented using an IDE; Notepad for the first project and an online IDE, repl.it for the other two. The trace ball project requires a mouse follow a shape, so within that project, I included an algorithm for having a shape follow the mouse via tracking coordinates of the mouse and the browser page. The high low card game has algorithms for betting, randomization of cards and guessing if the value is higher or lower. The third project has two algorithms for reading files and matching words to a user’s input. |

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

|  |
| --- |
| <https://github.com/LBruni98/Trace-Ball/wiki/Project-Wiki---Trace-Ball#debugging-process-and-debugging-facilities>  <https://github.com/LBruni98/High-Low-Card-Game#debugging>  <https://github.com/LBruni98/Anagram-Solver#debugging> |
| The links above, in my opinion, fulfil the criteria above. The first link is a project wiki of my first project which explains the debugging process with how it helps the program. All three links explain the debugging process for each project; the wiki contains the facilities of Notepad, which such facilities aren’t there and the other two with repl.it, which has most facilities seen in a common IDE. |

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

|  |
| --- |
| <https://github.com/LBruni98/High-Low-Card-Game/blob/master/main.cpp>  <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 needed to use. |

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

|  |
| --- |
| <https://github.com/LBruni98/Trace-Ball/wiki/Project-Wiki---Trace-Ball#process-of-implementation> |
| 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/Trace-Ball#ide>  <https://github.com/LBruni98/High-Low-Card-Game#ide>  <https://github.com/LBruni98/Anagram-Solver#ide> |
| 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#development-of-code> |
| 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/Trace-Ball/wiki/Project-Wiki---Trace-Ball#background> |
| The wiki 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#evaluation>  <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#development-of-code> |
| 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. |

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

|  |
| --- |
| <https://github.com/LBruni98/High-Low-Card-Game#ide> |
| The project here shows the how using the IDE has its benefits over the last one. Located within the debugging section showcases how easier it was to develop the program with repl.it. The comparison was with said IDE and notepad, detailing the difference of both. |

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

|  |
| --- |
| <https://github.com/LBruni98/High-Low-Card-Game#evaluation> |
| The link above shows suitable evidence of utilising a coding standard and how it helps. The link provided shows the project documentation, which within showcases a description of the coding standard and how it 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. |