# Task 2:

*Identify the user’s requirements for the above problem:*

The requirements for this program should be gathered in the initial stages, before development. Before releasing the product, interviews and feedback should be conducted.

Functional requirements should be first considered. Functional requirements constitute the program and how it should be structured. It consists of all the functions expected.

Functional requirements:

* The program requires their score
  + The user needs to be able to supply their score
* It will be important to investigate the exact implementation of the algorithm being used. What platform will it be used on? The functions used should cater for its provided platform (website for example)

Non-functional requirements describe how well the program will perform:

* Security measures
  + Contributors to the project are trusted
  + Use defensive coding practices if necessary
* The performance of the program is not much of a concern in this instance because nested loops are not being utilized. The program executes blocks of code with O(n) only.
* Variable naming convention should use Camel Case, and describe its functions clearly
* The code should be neatly structured for easy maintenance
* The Console prompts are to be clear, and describe the input requested, as well as the input constraints. They should not leave the user confused about how to operate the program

Constraints are describing the limitations of the program:

* The Program requires the correct constraints for each grade based on the score.
* The Program will also require constraints based on incorrect input that will prevent entries other than integers from 0-100
* The Program will be run locally from the console
* The user should be prompted of the input constraints

Acceptance Criteria describes the conditions that must be met:

* Based upon the requirements listed above, the program must display the correct mark respective to the score.
* It must efficiently handle incorrect inputs.
* It should be clear how to operate the program effectively.
* It should be verbose

# Task 4:

*a. Explore and tabulate five Programming languages along with the most popular  
applications of each of these languages.*

The vast array of computer programming languages empowers users to execute virtually any task computable. Each language possesses its own unique characteristics, expressed through its paradigm, syntax, and compilation or execution model. These distinct attributes influence a language's suitability for specific applications. Consequently, the selection of a programming language constitutes a crucial decision in the initial stages of software development.

Users choose a Programming Language based on a set of requirements defined in the initial stages of a project. These requirements can be multifaceted, often influenced by factors such as the user's industry, the project's scope, and its anticipated scale. Many programming languages exist for different purposes to satisfy these requirements.

Some examples of contrasting applications for the most popular programming languages today:

1:

|  |  |
| --- | --- |
| Language: | Applications: |
| Python | Scientific Computing, AI, Automation |
| **Python reigns supreme as the programming language of choice within numerous scientific disciplines.** Its extensive library ecosystem, particularly rich in Data Science and AI domains, empowers users  to tackle a remarkable breadth of problems. The intuitive syntax of Python further fuels its popularity, even extending its reach beyond traditional programmers. | |

2:

|  |  |
| --- | --- |
| Language: | Applications: |
| JavaScript | Client Side Web Development, Mobile Apps, User-Interfaces (Simplilearn, 2021) |
| **JavaScript powers the interactive and dynamic nature of modern websites.** This versatile scripting language empowers web developers to craft engaging user experiences by enabling dynamic features directly within the user's browser. JavaScript executes on the client-side, eliminating the need for constant communication with a server, and contributing to a more responsive and interactive website experience. Its ubiquitous presence, powering over 97% of websites (W3Techs, 2024), equips online businesses with a powerful tool to showcase their products and services in a captivating and interactive manner. | |

3:

|  |  |
| --- | --- |
| Language: | Applications: |
| C | Operating Systems, Kernels, Device Drivers, Application Software |
| **C, a foundational general-purpose programming language, emerged in the early 1970s.** Despite its relative maturity, C remains a cornerstone of modern computing systems. Notably, C forms the core of virtually all operating system kernels, including those used in Windows, UNIX-like systems (including Linux), and various embedded systems. In this context, C acts as a critical intermediary layer, translating high-level application programming instructions into low-level commands that the hardware can directly execute.  **Originally developed for the UNIX operating system by Dennis Ritchie,** C's influence extends beyond kernels. Historically, C served as a prevalent language for application development. However, the rise of higher-level languages with features like automatic memory management and object-oriented programming paradigms has led to a shift in preference for application development. Nevertheless, C's efficiency, control over hardware resources, and extensive library ecosystem continue to make it a valuable tool for system programming and performance-critical applications. | |

4:

|  |  |
| --- | --- |
| Language: | Applications: |
| Ruby | Web Development and Deployment, Web Scraping and Data Processing |
| Ruby's expressive syntax prioritizes developer productivity with its concise and readable structure. This focus on developer experience extends to web development, where Ruby boasts a robust ecosystem of powerful web libraries, often referred to as gems. Notably, the Rails framework streamlines web application development and deployment (webcrunch, 2024), making Ruby a top choice for building complex enterprise applications. While both Ruby and Python share strengths in web development, Ruby carves a niche in the enterprise sector, while Python reigns supreme in scientific computing and machine learning (Coursera, 2024). Highlighting its industry adoption, the ubiquitous Ruby on Rails framework powers numerous high-traffic web applications like GitHub and Twitch. | |

5:

|  |  |
| --- | --- |
| Language: | Applications: |
| C# | Web Services/Applications, Game Development, Application Software |
| C#, designed by Microsoft, emerged in response to the growing popularity of Java. Microsoft developed the .NET framework, a comprehensive platform utilizing C# as its primary language. Over time, C# has gained significant traction, narrowing the gap with Java in terms of usage (Javatpoint, n.d.-b). C# excels in corporate environments, particularly for large-scale projects like game development and scalable web applications. The .NET framework itself enjoys a positive reputation among programmers due to its robust features and ease of use. (Javatpoint, n.d.) | |

*b. Find an example of software used in business and report their use e.g., ATM  
machine or shopping centre pay machine. What sort of computer  
programming languages are used in such applications and who developed  
them?*

The Wikipedia website leverages the open-source software framework MediaWiki. This powerful platform, primarily written in PHP, utilizes JavaScript for client-side scripting functionalities. The Wikimedia Foundation, a non-profit organization, oversees MediaWiki's development. MediaWiki serves as the foundation for not only Wikipedia, but also numerous other wiki websites. These websites can adopt and adapt the MediaWiki codebase, a process often referred to as "forking," to suit their specific needs. (MediaWiki, 2012/2024)

**References:**

Coursera. (2024, June 4). *Ruby vs. Python: Pros, Cons, and Where to Start*. Coursera. https://www.coursera.org/articles/ruby-vs-python

Javatpoint. (n.d.-a). *C# Features*. Retrieved 30 June 2024, from https://www.javatpoint.com/csharp-features

Javatpoint. (n.d.-b). *C# History*. Retrieved 30 June 2024, from https://www.javatpoint.com/csharp-history

MediaWiki. (2024). *Wikimedia/mediawiki* [PHP]. Wikimedia. https://github.com/wikimedia/mediawiki (Original work published 2012)

Simplilearn. (2021, December 20). *10 Practical Applications of JavaScript And Tips | Simplilearn*. Simplilearn.Com. https://www.simplilearn.com/applications-of-javascript-article

W3Techs. (2024, June 27). *Usage Statistics of JavaScript as Client-side Programming Language on Websites, June 2024*. https://w3techs.com/technologies/details/cp-javascript/

webcrunch (Director). (2024, March 7). *Should you use Ruby on Rails in 2024?* https://www.youtube.com/watch?v=w7UzwO673Gg