Assignment

Design an algorithm that can generate a custom, reproducible password that is uniquely different for each website.

Working in pairs, your task is to design and construct a standardized strategy for generating unique passwords for different sites that can later be regenerated by reapplying the same algorithm. Your solution should address the following concepts:

- The algorithm should generate different passwords for different sites.
- The password for any site should be reproducible simply by following the algorithm.
- The algorithm should be easy to remember and apply.
- The password should be complex and difficult to guess.
- The general algorithm should not be easily deduced from the password.

Once you've designed your solution, write out each step of your password-generating algorithm in some form of *pseudocode*. No specific format is required for your algorithm, but your pseudocode should be clear enough and detailed enough that anyone who is not familiar with how your algorithm is supposed to work can still follow along and apply its steps in generating a valid password.

Submission

Your submission will be in the form of a written algorithm (i.e., pseudocode) that explicitly states each of the discrete steps and decisions that must be made in generating a valid password. Also, you must provide at least five examples of passwords that your algorithm would generate for five different sites. One of those examples must be thoroughly annotated, showing how each step of the algorithm contributes to the final password.

Your solution and examples should demonstrate the following properties:

- Clear and readable
- Cleanly formatted
- Appropriate use of sequencing, selection, and/or iteration
- Well-documented examples

Learning Goals

Over the course of this module and this project, you will learn to:

- identify and examine a number of common features of algorithms, including sequencing, selection, and repetition
- write pseudocode to describe each step of an algorithm with clarity and precision
- construct trace tables documenting the result of each step of an algorithm

- compare the differences between different types of common algorithms
- analyze the need for artificial programming languages
- examine strategies for approaching large-scale problems
- identify factors that allow solutions to scale efficiently
- encode and decode messages using common cryptographic techniques
- examine a number of common threats to cybersecurity, including distributed denial of service attacks (DDoS), phishing, viruses, and social engineering
- examine the implications of Moore's Law on the research and development of new and existing technologies

Rubric

Content Area	Performance Quality			
Readability	Algorithm is typed, organized, and nicely formatted for easy use.	Algorithm is organized and nicely formatted for easy use, but is not typed. **—OR—** Algorithm is typed, but the formatting and organization makes it somewhat difficult to use.	Algorithm has formatting and organization that makes it somewhat difficult to use **AND** is not typed. **—OR—** Algorithm may be typed, but the formatting and organization makes it extremely difficult to use.	Not enough criteria are met in order to award any credit.
Flow	The algorithm incorporates the appropriate use of all three types of programming structure: sequencing, selection, and iteration.	The algorithm incorporates the appropriate use of only two types of programming structure: sequencing, selection, and iteration.	The algorithm incorporates the appropriate use of only one type of programming structure: sequencing, selection, and iteration.	Not enough criteria are met in order to award any credit.

Correctness	The algorithm generates a unique and reproducible password for all sites.	The algorithm generates a reproducible password for all sites, however, some may not be unique. **—OR—** The algorithm generates a unique and reproducible password for most sites. **—OR—** The algorithm generates a unique password for all sites, however, it is not reproducible.	The algorithm generates a password for all sites, however, some may not be unique or reproducible. **—OR—** The algorithm generates a unique and reproducible password for only a few sites.	Not enough criteria are met in order to award any credit.
Effectiveness	The algorithm cannot be easily deduced from just the password and the name of the site.	A few parts of the algorithm can be easily deduced from just the password and the name of the site.	Most parts of the algorithm can be easily deduced from just the password and the name of the site.	Not enough criteria are met in order to award any credit.
Examples	There are five sample passwords generated correctly based on the algorithm.	There are four sample passwords generated correctly based on the algorithm.	There are three or fewer sample passwords generated correctly based on the	Not enough criteria are met in order to award any credit.

			algorithm.	
Documented Case	There is one annotated example documented at all steps of the process. **—AND—** It is well formatted and organized and easy to follow.	There is one annotated example documented at most steps of the process **AND** It is well formatted and organized and easy to follow. **—OR—** There is one annotated example documented at all steps of the process, but the organization and formatting makes it difficult to follow.	There is one annotated example documented at some steps of the process **AND** It is well formatted and organized and easy to follow. **—OR—** There is one annotated example documented at all steps of the process, but the organization and formatting makes it extremely difficult to follow. **—OR—** There is one annotated example documented at most steps of the process, but the organization and formatting make it difficult to follow.	Not enough criteria are met in order to award any credit.