CAMPAIGN CODE GENERATOR

Algorithm

The algorithm generates random codes by selecting characters from a provided pool and ensures uniqueness by removing duplicates. It continues generating codes until the desired count is reached, providing a simple and effective way to generate random codes of a specified length.

Checker

As requested in the expected section, the checker method does not search any data structure or database. It just fixes duplicates. This is a situation that provides the desired uniqueness.

Minimum Predictability

According to my research, The RNGCryptoServiceProvider class in .NET, used in the RNGCryptoServiceProviderAdapter, is considered to be more suitable for generating cryptographically secure random numbers compared to the Random class or other pseudorandom number generators.

Here are a few reasons why RNGCryptoServiceProvider is preferred for cryptographic applications:

- Cryptographic Security: RNGCryptoServiceProvider is designed to provide random numbers that are suitable for cryptographic use. It uses cryptographic algorithms and a secure entropy source to generate random bytes, ensuring a high level of unpredictability and security.
- True Randomness: RNGCryptoServiceProvider utilizes hardware entropy sources, such as thermal noise or other physical phenomena, to generate random numbers. This makes it suitable for generating truly random numbers, as opposed to pseudorandom numbers generated by algorithms like those used in the Random class.
- Cryptographic Applications: RNGCryptoServiceProvider is specifically designed for cryptographic purposes, such as generating encryption keys, initialization vectors, or nonces. These applications require a high degree of randomness and security, and RNGCryptoServiceProvider provides the necessary level of confidence.