

This repository [Explore](#) [Features](#) [Enterprise](#) [Blog](#)[Sign up](#)[Sign in](#) pvginkel / **ProgressEncode**[Star](#) 3[Fork](#) 1

Implementation of the Progress OpenEdge ENCODE algorithm

1 commit

1 branch

0 releases

Fetching contributors



branch: master ▾

ProgressEncode / 

Fetching latest commit...

[.nuget](#)[ProgressEncode.Test](#)[ProgressEncode](#)[.gitignore](#)[License.txt](#)[ProgressEncode.sln](#)[README.markdown](#) **Code**[Issues](#)

4

[Pull Requests](#) **Pulse**[Graphs](#)**HTTPS** clone URL<https://github.com/> You can clone with [HTTPS](#) or [Subversion](#).[7 Download ZIP](#)2 **README.markdown**

ProgressEncode

LGPL License.

Introduction

Progress OpenEdge 4GL has a function named `ENCODE`. This function is a one way hash function that is used by many legacy applications written in Progress to e.g. hash passwords.

Progress has stated (<http://knowledgebase.progress.com/articles/Article/P111508>) that they are not willing to make the algorithm behind the `ENCODE` function public. This can cause problems when you need to interoperate with legacy applications. This project contains a C# implementation of the algorithm behind the Progress `ENCODE` function specifically to allow this kind of operability.

Issues

The unit tests which are part of the project contain a randomly generated set of inputs and outputs encoded by Progress. The only known issue is that Progress does not allow `NUL` characters in strings, so I could not test this. The list of encoded strings in the unit tests all pass.

Bugs

Bugs should be reported through github at <http://github.com/pvginkel/ProgressEncode/issues>.

License

ProgressEncode is licensed under the LGPL 3.