Package ‘seguid’

March 4, 2024

Version 0.1.0

Title Sequence Globally Unique Identifier (SEGUID) Checksums

Description

Implementation of the original Sequence Globally Unique Identifier (SEGUID) algorithm [Babnigg and Giometti (2006) <doi:10.1002/pmic.200600032>] and SEGUID v2 (<https://www.seguid.org>), which extends SEGUID v1 with support for linear, circular, single- and double-stranded biological sequences, e.g. DNA, RNA, and proteins.

Imports base64enc, digest

License MIT + file LICENSE

URL https://www.seguid.org/

BugReports https://github.com/seguid/seguid-r/issues

Encoding UTF-8

RoxygenNote 7.3.1

NeedsCompilation no

Author Henrik Bengtsson [aut, cre, cph]

Maintainer Henrik Bengtsson <henrikb@braju.com>

Repository CRAN

Date/Publication 2024-03-04 10:40:02 UTC

R topics documented:

`seguid` *SEGUID checksums for linear, circular, single- and double-stranded sequences*

Description

SEGUID checksums for linear, circular, single- and double-stranded sequences

Usage

```
seguid(seq, alphabet = "{DNA}", form = c("long", "short", "both"))

lsseguid(seq, alphabet = "{DNA}", form = c("long", "short", "both"))

csseguid(seq, alphabet = "{DNA}", form = c("long", "short", "both"))

ldseguid(watson, crick, alphabet = "{DNA}", form = c("long", "short", "both"))

cdseguid(watson, crick, alphabet = "{DNA}", form = c("long", "short", "both"))
```

Arguments

`seq` (character string) The sequence for which the checksum should be calculated. The sequence may only comprise of symbols in the alphabet specified by the `alphabet` argument.

`alphabet` (character string) The type of sequence used. If `"{DNA}"` (default), then the input is a DNA sequence. If `"{RNA}"`, then the input is an RNA sequence. If `"{protein}"`, then the input is an amino-acid sequence. If `"{DNA-extended}"` or `"{RNA-extended}"`, then the input is a DNA or RNA sequence specified an extended set of symbols, including If `"{protein-extended}"`, then the input is an amino-acid sequence with an extended set of symbols, including IUPAC symbols (4). A custom alphabet may also be used. A non-complementary alphabet is specified as a comma-separated set of single symbols, e.g. `"X,Y,Z"`. A complementary alphabet is specified as a comma-separated set of paired symbols, e.g. `"AT,CG"`. It is also possible to extend a pre-defined alphabet, e.g. `"{DNA},XY"`.

`form` (character string) How the checksum is presented. If `"long"` (default), the full-length checksum is outputted. If `"short"`, the short, six-digit checksum is outputted. If `"both"`, both the short and the long checksums are outputted.

`watson, crick` (character strings) Two reverse-complementary DNA sequences. Both sequences should be specified in the 5’-to-3’ direction.

Value

The SEGUID functions return a single character string, if `form` is either `"long"` or `"short"`. If `form` is `"both"`, then a character vector of length two is return, where the first component holds the "short" checksum and the second the "long" checksum. The long checksum, without the prefix, is string with 27 characters. The short checksum, without the prefix, is the first six characters of the long checksum. All checksums are prefixed with a label indicating which SEGUID method was used. Except for `seguid()`, which uses *base64* encoding, all functions produce checksums using the *base64url* encoding ("Base 64 Encoding with URL and Filename Safe Alphabet").

`seguid()` calculates the SEGUID v1 checksum for a linear, single-stranded sequence.

`lsseguid()` calculates the SEGUID v2 checksum for a linear, single-stranded sequence.

`csseguid()` calculates the SEGUID v2 checksum for a circular, single-stranded sequence.

`ldseguid()` calculates the SEGUID v2 checksum for a linear, double-stranded sequence.

`cdseguid()` calculates the SEGUID v2 checksum for a circular, double-stranded sequence.

Base64 and Base64url encodings

The base64url encoding is the base64 encoding with non-URL-safe characters substituted with URL-safe ones. Specifically, the plus symbol (`+`) is replaced by the minus symbol (`-`), and the forward slash (`/`) is replaced by the underscore symbol (`_`).

The Base64 checksum, which is used for the original SEGUID checksum, is not guaranteed to comprise symbols that can safely be used as-is in Uniform Resource Locator (URL). Specifically, it may consist of forward slashes (`/`) and plus symbols (`+`), which are characters that carry special meaning in a URL. For the same reason, a Base64 checksum cannot safely be used as a file or directory name, because it may have a forward slash.

The checksum returned is always 27-character long. This is because the representation always end with a padding character (`=`) so that the length is a multiple of four character. We relax this requirement, by dropping the padding character.

References

1. Babnigg, G., Giometti, CS. A database of unique protein sequence identifiers for proteome studies. Proteomics. 2006 Aug;6(16):4514-22. doi:10.1002/pmic.200600032.
2. Josefsson, S., The Base16, Base32, and Base64 Data Encodings, RFC 4648, doi:10.17487/RFC4648, October 2006, `https://www.rfc-editor.org/info/rfc4648`.
3. Wikpedia article 'Nucleic acid notation', February 2024. `https://en.wikipedia.org/wiki/Nucleic_acid_notation`.
4. Wikpedia article 'Nucleic acid notation', February 2024, `https://en.wikipedia.org/wiki/Amino_acid`.
5. Wikipedia article 'SHA-1' (Secure Hash Algorithm 1), December 2023. `https://en.wikipedia.org/wiki/SHA-1`.

Examples

```
## SEGUID v1 on linear single-stranded DNA
seguid("GATTACA")
#> seguid=tp2jzeCM2e3W4yxtrrx09CMKa/8

## SEGUID v2 on linear single-stranded DNA
lsseguid("GATTACA")
```

```
#> lsseguid=tp2jzeCM2e3W4yxtrrx09CMKa_8

## SEGUID v2 on cicular single-stranded DNA
## GATTACA = ATTACAG = ... = AGATTAC
csseguid(”GATTACA”)
#> csseguid=mtrvbtuwr6_MoBxvtm4BEpv-jKQ

## SEGUID v2 on blunt, linear double-stranded DNA
##   GATTACA
##   CTAATGT
ldseguid(”GATTACA”, ”TGTAATC”)
#> ldseguid=AcRsEcNFrui5wCxI7xxo6wnDYPY

## SEGUID v2 on staggered, linear double-stranded DNA
##   -ATTACA
##   CTAAT--
ldseguid(”-ATTACA”, ”--TAATC”)
#> ldseguid=98Klwxd3ZQPGHqnH3BheIuZVHQQ

## SEGUID v2 on circular double-stranded DNA
## GATTACA = ATTACAG = ... = AGATTAC
## CTAATGT = TAATGTC = ... = TCTAATG
cdseguid(”GATTACA”, ”TGTAATC”)
#> cdseguid=zCuq031K3_-40pArbl-Y4N9RLnA

## SEGUID v2 on linear single-stranded expanded
## epigenetic sequence (Viner et al., 2024)
viner_DNA <- ”{DNA},m1,h2,f3,c4”
lsseguid(”AmT2C”, alphabet = viner_DNA)
#> lsseguid=MW4Rh3lGY2mhwteaSKh1-Kn2fGA

## SEGUID v2 on linear double-stranded expanded
## epigenetic sequence (Viner et al., 2024)
ldseguid(”AmT2C”, ”GhA1T”, alphabet = viner_DNA)
#> ldseguid=rsPDjP4SWr3-ploCeXTdTA80u0Y
```

Index