T:mk.tmp

COLLABORATORS

	TITLE : T:mk.tmp	

ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		July 1, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

T:mk.tmp

1.1 main

```
--background--

rexx_stem.library/StemCopy

rexx_stem.library/StemInsert

rexx_stem.library/StemRead

rexx_stem.library/StemRemove

rexx_stem.library/StemSearch

rexx_stem.library/StemSort

rexx_stem.library/StemWrite
```

1.2 --background--

```
--background-- :


 $(C): (1996, Rocco Coluccelli, Bologna)
 $VER: rexx_stem.library 37.3 (15.03.97)

    rexx_stem.library

    This sub-library of the rexxMOOS.library let ARexx programmers
    perform very powerfull manipulation over stem and compound symbols.

        StemCopy()
        StemInsert()
        StemRead()
        StemRemove()
        StemSearch()
```

```
        StemSort()
        StemWrite()

    NOTES

        Is part of the MOOS package.

    TODO

        StemToString()

    BUGS
```

1.3 rexx_stem.library/StemCopy

```
StemCopy :


    NAME

        StemCopy -- Copy a stem into another.

    SYNOPSIS

        success = StemCopy(options)

    FUNCTION

        StemCopy() take a source stem and copy its entries into a
        destination stem. By default the source stem will be appended
        at the end of the destination.

    INPUTS

        options - "FromStem/A,ToStem/A,FromPos/K/N,ToPos/K/N,Tot/K/N"

            "FromStem" - The source stem.

            "ToStem"   - The destination stem, will receive entries
                         from the source stem. Any type of compound
                         symbol ("#", "*", "**") is supported from
                         this function (read the rexxMOOS documentation
                         for details) and this is valid for "FromStem"
                         too.

            "FromPos"  - First entry, in the source stem, to be
                         copied. (default is 0)

            "ToPos"    - Position where start to copy all entries.
                         The existing entries will be overwritten.
                         (default is last entry, append source stem
                         to destination)

            "Tot"      - How many entries, from the source stem,
                         need to copy into the destination.
```

```
                        (default is all entries)

    RESULT

        success - boolean value (1 mean all done, ok).

    EXAMPLE

        from. = "From stem...."; from.count = 5; to.count = 12
        DO i = 0 FOR to.count
            to.i.post = "to stem" i
        END
        IF StemCopy('from. to.#.post ToPos 2') THEN
            DO i = 0 FOR to.count
                SAY "TO."i".POST ==" to.i.post
            END

    NOTES

    BUGS

    SEE ALSO

        rexx_stem.library/StemInsert()
```

1.4 rexx_stem.library/StemInsert

```
StemInsert :


    NAME

        StemInsert -- Insert a stem into another.

    SYNOPSIS

        success = StemInsert(from,options)

    FUNCTION

        StemInsert() take a source stem and insert its entries into a
        destination stem.

    INPUTS

        from    - The source stem.

        options - "ToStem/A,FromPos/K/N,ToPos/K/N,Tot/K/N"

            "ToStem"  - The destination stem, will receive entries from
                        the source stem. Any type of compound symbol
                        ("#", "*", "**") is supported from this function
                        (read the rexxMOOS documentation for details).

            "FromPos" - First entry, in the source stem, to be
```

```
                        inserted. (default is 0)

            "ToPos"   - Position where start to insert all entries.
                        The existing entries will be shifted down
                        before inserting the others from the source.
                        (default is after the last entry of the
                        destination stem)

            "Tot"     - How many entries, from the source stem,
                        need to insert into the destination.
                        (default is all entries)

    RESULT

        success - boolean value (1 mean all done, ok).

    EXAMPLE

        in. = "empty"; in.count = 6
        DO i = 0 FOR in.count; SAY in.i; END
        IF StemInsert('in.','out. FromPos 3 Tot 4') THEN
            DO i = 0 FOR out.count
                SAY out.i
            END

    NOTES

    BUGS

    SEE ALSO
```

1.5 rexx_stem.library/StemRead

```
StemRead :


    NAME

        StemRead -- Read a file putting its lines into a stem.

    SYNOPSIS

        success = StemRead(filepath,options)

    FUNCTION

        The function read the given file line by line and put
        them into a specified stem.

    INPUTS

        filepath - Relative or absolute path. This argument must
                   be provided and the file must exists.

        options  - "OutStem/A,Del=Delete/S"
```

```
            "OutStem" - The stem to put readed lines in.
                        Any type of compound symbol ("#", "*", "**")
                        is supported from this function (read the
                        rexxMOOS documentation for details).

            "Delete"  - Delete the file after read.

    RESULT

        success - boolean value (1 mean all done, ok).

    EXAMPLE

        IF StemRead("S:User-Startup","txt.") THEN
            DO n = 0 FOR txt.count
                SAY txt.n
            END

    NOTES

        The maximum length for a line must be less than 8192 characters.

    SEE ALSO

        rexx_stem.library/StemWrite()
```

1.6 rexx_stem.library/StemRemove

```
StemRemove :


    NAME

        StemRemove -- Remove a block of entries.

    SYNOPSIS

        success = StemRemove(options)

    FUNCTION

        This function take a stem and remove a block of entries.
        The stem will be repacked updating also the dimension (count).

    INPUTS

        options - "InStem/A,FromPos/K/N,ToPos/K/N,Tot/K/N"

            "InStem"  - The stem to perform removing on.
                        Any type of compound symbol ("#", "*", "**")
                        is supported from this function (read the
                        rexxMOOS documentation for details).

            "FromPos" - The first entry to be removed.
```

```
                            (default is 0)

          "ToPos"    - The last entry to be removed. All other
                       entries, from ToPos+1 to the last, will
                       take place over the deleted ones.
                       (default is last entry of the stem)

          "Tot"      - Instead of using ToPos you can specify
                       how many entries need to remove.
                       (default is all entries)

    RESULT

        success - boolean value (1 mean all done, ok).

    EXAMPLE

        in. = "empty"; in.2 = "full"; in.4 = "full"; in.count = 10
        DO i = 0 FOR in.count; SAY in.i; END
        IF StemRemove('in. FromPos 3 Tot 5') THEN
            DO i = 0 FOR in.count
                SAY in.i
            END

    NOTES

        Using this function to remove ALL entries from a stem is
        a non-sense, use the instruction DROP instead.

    BUGS

    SEE ALSO

        rexx_stem.library/StemInsert()
```

1.7 rexx_stem.library/StemSearch

```
StemSearch :


    NAME

        StemSearch -- Search a pattern into the given stem.

    SYNOPSIS

        success = StemSearch(pattern,options)

    FUNCTION

        StemSearch() do a pattern matching over entries of a given stem.
        It will fill an output stem with all found entries.

    INPUTS
```

```
        pattern - The pattern to search for (standard AmigaDOS).

        options - "InStem/A,OutStem/A,Pos/K/N,Case/S"

            "InStem"  - Stem where perform the pattern matching.
                        Any type of compound symbol ("#", "*", "**")
                        is supported from this function (read the
                        rexxMOOS documentation for details). This is
                        valid also for "OutStem".

            "OutStem" - Output stem for all matched entries. Any type
                        of compound symbol is supported.

            "Pos"     - Start searching from the entry at "Pos"...
                        (default is the first entry)

            "Case"    - Do search case sensitive.
                        (default NoCase)

    RESULT

        success - boolean value (1 mean all done, ok).

    EXAMPLE

        in. = "empty"; in.2 = "String"; in.5 = "string"
        in.count = 10
        IF StemSearch("st#?","in. out.") THEN
            DO n = 0 FOR out.count; SAY out.n; END

    NOTES

    BUGS

    SEE ALSO

        dos.library/MatchPattern, dos.library/MatchPatternNoCase
```

1.8 rexx_stem.library/StemSort

```
StemSort :


    NAME

        StemSort -- Sort entries into the given stem.

    SYNOPSIS

        success = StemSort(options)

    FUNCTION

        StemSort() perform a sorting of entries found into a stem.
        It can fill an output stem with all sorted entries. The sort
```

```
        function is, by default, case insensitive.

    INPUTS

        options - "InStem/A,OutStem,Descend/S,NoDupes/S,Case/S"

            "InStem"  - Stem to be sorted. All entries will be
                        overwritten by the sorted ones.
                        Any type of compound symbol ("#", "*", "**")
                        is supported from this function (read the
                        rexxMOOS documentation for details).

            "OutStem" - Optionally you can give an output stem
                        where store all sorted entries leaving
                        the InStem untouched.

            "Descend" - Sort in alphabetical descend order (default
                        is "Ascend").

            "NoDupes" - Remove duplicates entries.

            "Case"    - Force sort to be case sensitive.

    RESULT

        success - boolean value (1 mean all done, ok).

    EXAMPLE

        in.0 = 20
        DO n = 1 FOR in.0; in.n = ERRORTEXT(n); SAY in.n; END
        IF StemSort("in.* out.") THEN
            DO n = 0 FOR out.count
                SAY out.n
            END

    NOTES

        Use the tqsort() function to sort case sensitive and a built-in
        Tqsort for sorting case insensitive.

    SEE ALSO
```

1.9 rexx_stem.library/StemWrite

```
StemWrite :


    NAME

        StemWrite -- Write the given stem into a file.

    SYNOPSIS

        success = StemWrite(filepath,options)
```

```
FUNCTION

    The function write entries from a given stem to a file.

INPUTS

    filepath - Relative or absolute path. This argument must
               be provided.

    options  - "InStem/A,Append/S"

        "InStem" - The source stem to take entries from.
                   Any type of compound symbol ("#", "*", "**")
                   is supported from this function (read the
                   rexxMOOS documentation for details).

        "Append" - Don't overwrite the output file if exists.

RESULT

    success - boolean value (1 mean all done, ok).

EXAMPLE

    txt. = "Prova StemWrite()"; txt.count = 20
    IF StemWrite("T:prova","txt.") THEN
        SAY "All done..."

NOTES

SEE ALSO

    rexx_stem.library/StemRead()
```