Text searching and sorting routines in Igor do not do any form of Unicode normalization. As a consequence, searching for the precomposed form of small letter n with tilde (U+00F1) in a string that contains the decomposed form (U+006E U+0303) will not result in a match. To get the desired result, you would need to first pass both the target string and the string to be searched through NormalizeUnicode using the same value for the *normalizationForm* parameter.

Example

```
Function TestNormalizeUnicode()
    String precomposed = "Ni" + "\u00F1" + "o"
    String decomposed = "Ni" + "\u006E\u0303" + "o"
    String precomposedTarget = "\u00F1"
    String decomposedTarget = "\u006E\u0303"
    Variable foundPos

    // SUCCESSFUL TESTS
    // Searching the precomposed string for the precomposed target is successful.
    foundPos = strsearch(precomposed, precomposedTarget, 0)
    Print foundPos                // Prints 2

    // Likewise, searching the decomposed string for the decomposed target is successful.
    foundPos = strsearch(decomposed, decomposedTarget, 0)
    Print foundPos                // Prints 2

    // UNSUCCESSFUL TESTS
    // Searching the precomposed string for the decomposed target fails.
    foundPos = strsearch(precomposed, decomposedTarget, 0)
    Print foundPos                // Prints -1

    // Likewise, searching the decomposed string for the precomposed target fails.
    foundPos = strsearch(decomposed, precomposedTarget, 0)
    Print foundPos                // Prints -1

    // USING NormalizeUnicode() FUNCTION
    Variable normForm = 2         // Could use 0-3 and the results would be the same.

    String precomposedNorm = NormalizeUnicode(precomposed, normForm)
    String decomposedNorm = NormalizeUnicode(decomposed, normForm)
    String precomposedTargetNorm = NormalizeUnicode(precomposedTarget, normForm)
    String decomposedTargetNorm = NormalizeUnicode(decomposedTarget, normForm)

    // Now, searching either precomposedNorm or decomposedNorm for either
    // precomposedTargetNorm or decomposedTargetNorm will give a match.
    Print strsearch(precomposedNorm, precomposedTargetNorm, 0)  // Prints 2
    Print strsearch(decomposedNorm, precomposedTargetNorm, 0)   // Prints 2
    Print strsearch(precomposedNorm, decomposedTargetNorm, 0)   // Prints 2
    Print strsearch(decomposedNorm, decomposedTargetNorm, 0)    // Prints 2
End
```

See Also

Text Encodings on page III-459, **String Variable Text Encoding Error Example** on page III-479

http://en.wikipedia.org/wiki/Unicode_equivalence

http://unicode.org/reports/tr15/#Norm_Forms

note

note(*waveName*)

The note function returns a string containing the note associated with the specified wave.

See Also

To create a wave note, use the **Note** *operation*.

Note

Note [**/K/NOCR**] ***waveName*** [**,** ***str***]

The Note operation appends *str* to the wave note for the named wave.

Parameters

str is a string expression.