Unicode:

When nothing is anything else

SEIU
DATA SYSTEMS
& ANALYTICS

Inspired by:

Marie Roald &
Yngve Mardal

Matching: Sometimes you get lucky

```
>>> name1 = "David"
>>> name2 = "david"
>>> name3 = "DAVID"
>>> name1 == name2
False
>>> name1.lower() == name2.lower()
True
>>> name2.upper() == name3.upper()
True
```

Matching: Sometimes it gets harder

```
>>> road1 = "Straße"
>>> road2 = "Strasse"
>>> road1 == road2
False
>>> road1.lower() == road2.lower()
False
>>> road1.upper() == road2.upper()
True
```

Matching: Being more careful

```
>>> r3 = "straße"
>>> r4 = "STRAßE"
>>> r3.upper() == r4.upper()
False
>>> r3.casefold() == r4.casefold()
True
```

Matching: Normalization still matters

```
>>> hot1 = b'jalape\xc3\xb1o'.decode()
>>> hot2 = b'jalapen\xcc\x83o'.decode()
>>> hot1, hot2
('jalapeño', 'jalapeño')
>>> hot1.casefold() == hot2.casefold()
False
>>> from unicodedata import normalize
>>> normalize("NFC", hot1) ==
    normalize("NFC", hot2)
True
```

Matching: Canonical encoding

```
>>> work1 = "office"
>>> work2 = "office"
>>> normalize("NFC", work1) ==
    normalize("NFC", work2)
False
>>> normalize("NFD", work1) ==
    normalize("NFD", work2)
False
```

Matching: Compatibility encoding

```
>>> work1 = "office"
>>> work2 = "office"
>>> normalize("NFKC", work1) ==
    normalize("NFKC", work2)
True
>>> normalize("NFKD", work1) ==
    normalize("NFKD", work2)
True
```

Matching: A Byzantine conspiracy

```
>>> constantinople1 = "İstanbul"
>>> constantinople2 = "istanbul"
>>> constantinople1 == constantinople2
False
>>> normalize("NFKC", constantinople1)
      .casefold() ==
    normalize("NFKC", constantinople2)
      .casefold()
False
```

Matching: Unicode has no locale

```
def turkish_delight(s1, s2):
    s1 = normalize("NFKC", s1)
    s2 = normalize("NFKC", s2)
    s1 = (s1.replace("İ", "<dotted>")
          .replace("i", "<dotted>")
          .replace("I", "<nodot>")
          .replace("ı", "<nodot>"))
    s2 = (s2.replace("İ", "<dotted>")
          .replace("i", "<dotted>")
          .replace("I", "<nodot>")
          .replace("ı", "<nodot>"))
    return s1 == s2
```

Matching: Stylish numeracy

```
>> ns1, ns2 = "¹²³⁴", "1234"
>>> ns1.casefold() == ns2.casefold()
False
>>> normalize("NFC", ns1) ==
    normalize("NFC", ns2)
False
>>> normalize("NFKC", ns1) ==
    normalize("NFKC", ns2)
True
```

Matching: Georgia on my mind

```
>>> a1, a2, a3, a4 = "Ⴀ⃞ა⃞" # Kartvelian Ani
>>> a1.casefold() == a2.casefold()
True
>>> a3.casefold() == a4.casefold()
True
>>> a1.casefold() == a4.casefold()
False
>>> for c in a1, a2, a3, a4: print(name(c))
GEORGIAN CAPITAL LETTER AN
GEORGIAN SMALL LETTER AN
GEORGIAN LETTER AN
GEORGIAN MTAVRULI CAPITAL LETTER AN
```