

Esercizio 1

Con il termine “slug” si intende una stringa in formato leggibile in cui alcuni caratteri difficili da stampare vengono convertiti in altri facili da leggere.

Un possibile algoritmo è il seguente:

- tutti i caratteri accentati vengono convertiti nell'equivalente non accentato
- tutto viene convertito in minuscolo
- ogni carattere che non sia in [a-z][0-9] viene convertito in “-”
- due “-” consecutivi non sono ammessi, solo il primo viene tenuto
- un “-” finale non è ammesso a meno che non sia l'unico carattere nella stringa

L'obiettivo dell'esercizio è fare un funzione "slugify" che converta una stringa generica in uno slug.

Creare con cargo un nuovo package chiamato **slufigy** e, al suo interno, definire la funzione

```
fn slugify(s: &str) -> String {}
```

Per convertire le lettere accentate usare questa tabella di conversione, dove il carattere nella posizione i (come carattere) in SUBS I corrisponde al carattere nella posizione i in

SUBS O:

```
const SUBS_I : &str =  
    "àáâãäåæçèéêëëēğġñîïíìĩłľṁńňŋħōöøœϑθðóþřŗśšşșțȚüûũúűůųẂẄ  
ÿýžźż";  
  
const SUBS_O: &str =  
    "aaaaaaaaaacccddeeeeeeeegghiiiiiiilmmnnnooooooooooprsssstttuuuuuuuu  
uwxyyyzzz";
```

Per la conversione definire una funzione che esegua la conversione con la tabella definita;

```
fn conv(c: char) -> char {}
```

Nel main leggere una sequenza di parole come argomento da command line, invocare la funzione e stampare il risultato.

Esempio:

```
cargo run -- "Questo sarà uno slug!"
```

Attenzione, quando si esegue la conversione delle stringhe accentate: si può usare il risultato di find() dentro SUBS I come indice di SUBS O?

Suggerimento: la lunghezza di un char è sempre uguale?

Scrivere anche i test case necessari per verificare il corretto comportamento. Coprire almeno i seguenti casi:

- stringa con più di una parola
- stringa con caratteri accentati
- stringa vuota
- stringa con più spazi e caratteri non validi
- stringa con solo caratteri non validi

Per il parsing degli argomenti di command line si suggerisce di inserire nel progetto la libreria clap

<https://docs.rs/clap/latest/clap/>