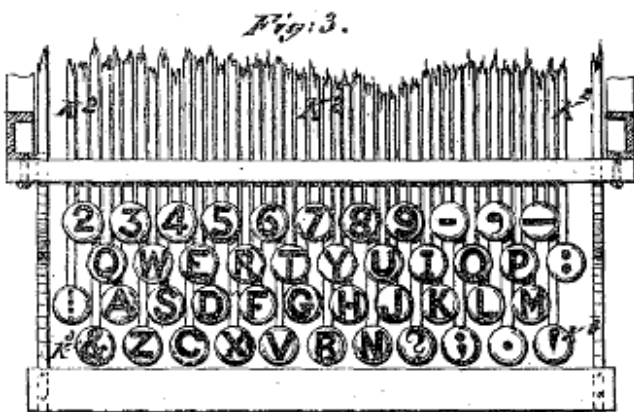


Keyboard

The inventor of the typewriter — [Christopher Scholes](#) — put the letters on his first model in alphabetical order. Because the symbols that are often used stood next to each other, the mechanical letter staves of the typewriter often bumped against each other and got stuck. In order to prevent this, Scholes reorganized the symbols of the keyboard. The first [Scholes and Glidden typewriter](#) that was eventually taken into production around 1873 at the company Remington already had a precursor of the current QWERTY-division. According to a popular myth, Remington placed the letters on the first row of the typewriter so that the sellers could make an impression on the customers by typing the product name TYPEWRITER using the symbols on the first row of the keyboard. It is also often said that TYPEWRITER is the longest word that can be typed with the letters of one single row of the keyboard. Reminders of the original alphabetical division of the keyboard can be found in the sequence DFGHJKL in the middle row of a keyboard.



QWERTY- keyboard division from a patent of Christopher Scholes (U.S. Patent No. 207,559 from 1878). Some keys are clearly the same as the modern QWERTY division: the C and the X changed places and the M is now next to the N.

Apart from QWERTY other divisions were developed, such as the AZERTY keyboards that are used in France and in Belgium, QWERTZ keyboards that are used in German-speaking and some Eastern-European countries, or the Dvorak keyboard, that was invented by August Dvorak because it is supposed to be better from an ergonomic point of view.

Assignment

1. Write a function `rowKey` which can be used to determine on which row of the keyboard a given symbol can be found. This symbol must be given to the function as an argument. The function must use the standard division of a QWERTY keyboard, but through an optional argument `keyboard`, another division can be given to the function. A keyboard division is described by a string where the symbols of the keyboard are listed from left to right and from top to bottom. Between two rows of a keyboard a vertical line (`|`) is placed in the string. The division of a QWERTY keyboard, for example, is given as `QWERTYUIOP|ASDFGHJKL|ZXCVBNM` and the division of an AZERTY keyboard as `AZERTYUIOP|QSDFGHJKLM|WXCVBN`. The first row is indicated with row number 1, and the function must print the value `None` if the given symbol isn't found on the keyboard. When searching, no distinction may be made between uppercase and lowercase letters.
2. Use the function `rowKey` to write a function `rowWord` with two parameters: an obligatory parameter `word` to which a given word must be given as an argument, and an optional

parameter `keyboard` with the same meaning as in the function `rowKey`. If the given word can be typed entirely with the letters of one single row of the keyboard, the function must print the number of that row (rows are again numbered from 1) as a result. Otherwise, the function should print the value `None`. Even if the word is an empty string, the value `None` must be printed. The function cannot make a distinction between uppercase and lowercase letters.

3. Use the function `rowWord` to write a function `longestWord`. To this function, a list of words must be given as an obligatory argument. From this word list, the function should print the longest word that can be typed using one row of the keyboard. If multiple words have an equal length, the function must print the first word that occurs in the word list. Next to a list of words, two optional parameters can be given to the function: an optional parameter `keyboard` with the same meaning as for the function `rowKey`, and an optional parameter `row` that specifically indicates the letters of which row of the keyboard the words can be formed with. If this last parameter isn't given, it doesn't matter which row the keyboard uses to write the words, as long as the words can be formed with the letters of one single keyboard row.

Example

```
>>> rowKey('U')
```

```
1
```

```
>>> rowKey('A')
```

```
2
```

```
>>> rowKey('z')
```

```
3
```

```
>>> rowKey('?')
```

```
>>> azerty = 'AZERTYUIOP|QSDFGHJKLM|WXCVCBN'
```

```
>>> rowKey('A', keyboard=azerty)
```

```
1
```

```
>>> rowWord('Tiruppur')
```

```
1
```

```
>>> rowWord('Jalajala')
```

```
2
```

```
>>> rowWord('Bavikove')
```

```
>>> azerty = 'AZERTYUIOP|QSDFGHJKLM|WXCVCBN'
```

```
>>> rowWord('Erattupetta', keyboard=azerty)
```

```
1
```

```
>>> rowWord('Bavikove', keyboard=azerty)
```

```
>>> hcesar = 'HCESAROPZ|QTDINULMX|YJBFVGKW'
```

```
>>> rowWord('Hazorasp', keyboard=hcesar)
```

```
1
```

```
>>> rowWord('Xiulin', keyboard=hcesar)
```

```
2
```

```
>>> rowWord('Bvkv', keyboard=hcesar)
```

```
3
```

```
>>> cities = ['Alaghsas', 'Erattupetta', 'Hazorasp', 'Piripiri', 'Tuntum']
```

```
>>> longestWord(cities)
```

```
'Alaghsas'
```

```
>>> longestWord(cities, row=1)
```

```
'Piripiri'
```

```
>>> longestWord(cities, row=2)
```

```

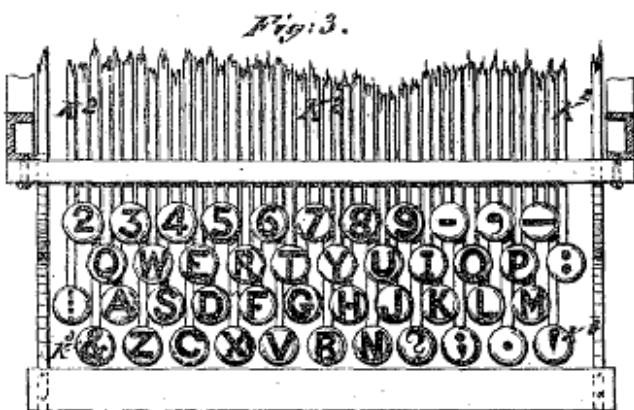
'Alaghsas'
>>> longestWord(cities, row=3)

>>> azerty = 'AZERTYUIOP|QSDFGHJKLM|WXCVBVN'
>>> longestWord(cities, keyboard=azerty)
'Erattupetta'
>>> longestWord(cities, keyboard=azerty, row=1)
'Erattupetta'
>>> longestWord(cities, row=2, toetsenbord=azerty)
>>> longestWord(cities, keyboard=azerty, row=3)

>>> hcesar = 'HCESAROPZ|QTDINULMX|YJBFVGKW'
>>> longestWord(cities, keyboard=hcesar)
'Hazorasp'
>>> longestWord(cities, keyboard=hcesar, row=1)
'Hazorasp'
>>> longestWord(cities, row=2, keyboard=hcesar)
'Tuntum'
>>> longestWord(cities, keyboard=hcesar, row=3)

```

De uitvinder van de schrijfmachine — [Christopher Scholes](#) — zette op zijn eerste model de letters op het toetsenbord in alfabetische volgorde. Omdat enkele veelgebruikte symbolen hierdoor naast elkaar kwamen te liggen, botsten de mechanische letterstangetjes van de schrijfmachine vaak tegen elkaar en bleven ze klem zitten. Om dit te verhelpen, herschikte Scholes de symbolen op het toetsenbord. De eerste [Scholes en Glidden schrijfmachine](#) die uiteindelijk rond 1873 in productie werd genomen bij het bedrijf Remington had reeds een voorloper van de hedendaagse QWERTY-indeling. Volgens een populaire mythe plaatste Remington de letters op de eerste rij van de schrijfmachine zodat verkopers de klanten konden imponeren door de productnaam TYPEWRITER uit te schrijven aan de hand van de symbolen op de eerste rij van het toetsenbord. Er wordt ook nog vaak beweerd dat TYPEWRITER het langste woord is dat kan geschreven worden met de letters van één enkele rij op een toetsenbord. Restanten van de oorspronkelijke alfabetische indeling van het toetsenbord zijn nog terug te vinden in de reeks DFGHJKL op de middelste rij van een toetsenbord.



QWERTY-toetsenbordindeling uit een patent van Christopher Sholes (U.S. Patent No. 207,559 uit 1878). Enkele toetsen zijn hier nog niet hetzelfde als bij de moderne QWERTY-indeling: de C en de X zijn omgewisseld en de M zit nu naast de N.

Naast QWERTY werden ook nog enkele andere toetsenbordindelingen ontwikkeld, zoals AZERTY-toetsenborden gebruikt in Frankrijk en België, QWERTZ-toetsenborden gebruikt in Duitstalige en een aantal Oost-Europese landen, of het Dvorak toetsenbord, bedacht door August Dvorak omdat het ergonomisch beter zou zijn.

Opgave

1. Schrijf een functie `rijToets` waarmee kan bepaald worden op welke rij van een toetsenbord een gegeven symbool kan teruggevonden worden. Dit symbool moet als argument aan de functie doorgegeven worden. De functie moet standaard de indeling van een QWERTY-toetsenbord gebruiken, maar via een optioneel argument `toetsenbord` kan aan de functie een andere indeling doorgegeven worden. Een toetsenbordindeling wordt omschreven door een string waarin de symbolen van het toetsenbord worden opgelijst van links naar rechts, en van boven naar onder. Tussen twee rijen van een toetsenbord wordt in de string telkens een verticale streep (`|`) geplaatst. De indeling van een QWERTY-toetsenbord wordt bijvoorbeeld weergegeven als `QWERTYUIOP|ASDFGHJKL|ZXCVBNM` en de indeling van een AZERTY-toetsenbord als `AZERTYUIOP|QSDFGHJKLM|WXCVCBN`. De eerste rij wordt aangegeven met rijnummer 1, en de functie moet de waarde `None` teruggeven indien het gegeven symbool niet op het toetsenbord wordt teruggevonden. Bij het opzoeken mag geen onderscheid gemaakt worden tussen hoofdletters en kleine letters.
2. Gebruik de functie `rijToets` om een functie `rijWoord` te schrijven met twee parameters: een verplichte parameter `woord` waaraan een gegeven woord als argument moet doorgegeven worden, en een optionele parameter `toetsenbord` met dezelfde betekenis als bij de functie `rijToets`. Indien het gegeven woord volledig kan uitgeschreven worden met de letters op één enkele rij van het toetsenbord, dan moet de functie het nummer van die rij (rijen worden opnieuw genummerd vanaf 1) als resultaat teruggeven. Anders moet de functie de waarde `None` teruggeven. Ook als het woord de lege string is, moet de waarde `None` teruggegeven worden. De functie mag geen onderscheid maken tussen hoofdletters en kleine letters.
3. Gebruik de functie `rijWoord` om een functie `langsteWoord` te schrijven. Aan deze functie moet een lijst van woorden als verplicht argument doorgegeven worden. Uit deze woordenlijst moet de functie het langste woord teruggeven dat kan geschreven worden met de letters op één enkele rij van het toetsenbord. Indien er meerdere zulke woorden zijn met maximale lengte, dan moet de functie het woord teruggeven dat als eerst voorkomt in de woordenlijst. Naast een lijst van woorden kunnen aan de functie nog twee optionele parameters doorgegeven worden: een optionele parameter `toetsenbord` met dezelfde betekenis als bij de functie `rijToets`, en een optionele parameter `rij` die specifiek aangeeft met de letters van welke rij van het toetsenbord de woorden moeten kunnen gevormd worden. Indien deze laatste parameter niet wordt doorgegeven, dan maakt het niet uit welke rij van het toetsenbord gebruikt wordt om de woorden uit te schrijven, zolang de woorden maar kunnen gevormd worden met de letters van één enkele rij van het toetsenbord.

Voorbeeld

```
>>> rijToets('U')
1
>>> rijToets('A')
2
>>> rijToets('z')
3
>>> rijToets('?')

>>> azerty = 'AZERTYUIOP|QSDFGHJKLM|WXCVCBN'
>>> rijToets('A', toetsenbord=azerty)
1

>>> rijWoord('Tiruppur')
```

```
1
>>> rijWoord('Jalajala')
2
>>> rijWoord('Bavikove')

>>> azerty = 'AZERTYUIOP|QSDFGHJKLM|WXCVCBN'
>>> rijWoord('Erattupetta', toetsenbord=azerty)
1
>>> rijWoord('Bavikove', toetsenbord=azerty)

>>> hcesar = 'HCESAROPZ|QTDINULMX|YJBFVGKW'
>>> rijWoord('Hazorasp', toetsenbord=hcesar)
1
>>> rijWoord('Xiulin', toetsenbord=hcesar)
2
>>> rijWoord('Bvkv', toetsenbord=hcesar)
3

>>> steden = ['Alaghsas', 'Erattupetta', 'Hazorasp', 'Piripiri', 'Tuntum']

>>> langsteWoord(steden)
'Alaghsas'

>>> langsteWoord(steden, rij=1)
'Piripiri'
>>> langsteWoord(steden, rij=2)
'Alaghsas'
>>> langsteWoord(steden, rij=3)

>>> azerty = 'AZERTYUIOP|QSDFGHJKLM|WXCVCBN'
>>> langsteWoord(steden, toetsenbord=azerty)
'Erattupetta'
>>> langsteWoord(steden, toetsenbord=azerty, rij=1)
'Erattupetta'
>>> langsteWoord(steden, rij=2, toetsenbord=azerty)
>>> langsteWoord(steden, toetsenbord=azerty, rij=3)

>>> hcesar = 'HCESAROPZ|QTDINULMX|YJBFVGKW'
>>> langsteWoord(steden, toetsenbord=hcesar)
'Hazorasp'
>>> langsteWoord(steden, toetsenbord=hcesar, rij=1)
'Hazorasp'
>>> langsteWoord(steden, rij=2, toetsenbord=hcesar)
'Tuntum'
>>> langsteWoord(steden, toetsenbord=hcesar, rij=3)
```