

Problembeschreibung In dieser Aufgabe sollen mittels Breitensuche kürzeste Wege in einem Labyrinth gefunden werden. Das gegebene Labyrinth ist in quadratische Felder unterteilt, die jeweils entweder passierbar ('P') oder unpassierbar ('U') sind. Von jedem passierbaren Feld können in einem Schritt die (maximal vier) horizontal oder vertikal angrenzenden Felder erreicht werden, sofern sie ebenfalls passierbar sind. Gesucht ist die minimale Anzahl an Schritten, die nötig ist, um vom Startfeld zum Zielfeld zu gelangen.

Das Paar (i,j) bezeichnet das Feld in der i -ten Zeile und der j -ten Spalte, wobei jeweils mit 0 begonnen wird. Im folgenden Beispiel mit dem Startfeld $(0,9)$ und Zielfeld $(2,2)$ wäre die richtige Antwort also 11 (das Startfeld wird nicht mitgezählt).

```
UPPPUPPPUP
PUUPPPUUUP
PPPUUU PPPP
UUPPPPPUUP
```

Schreiben Sie eine Funktion `abstand(s, t, dateiname="labyrinth.dat")`, die ein Labyrinth aus der Datei mit angegebenem Namen liest und die Anzahl der Schritte von $(s[0], s[1])$ nach $(t[0], t[1])$ in diesem Labyrinth zurückgibt. Wird kein Dateiname angegeben, so wird aus der Datei mit Namen `labyrinth.dat` im aktuellen Arbeitsverzeichnis gelesen.

Eingabedatei Jede nichtleere Zeile der Eingabedatei enthält gleich viele Zeichen aus der Menge $\{P,U\}$, die das Labyrinth auf die oben beschriebene Art darstellen, und wird durch ein Newline-Zeichen (`'\n'`) abgeschlossen. Somit endet die Eingabedatei mit einer leeren Zeile.

Aufrufparameter Die Funktion `abstand` hat als obligatorische Parameter zwei Paare $s = (s_0, s_1)$, $t = (t_0, t_1)$ natürlicher Zahlen, die den Start- und Endpunkt angeben, und als optionalen Parameter `dateiname` einen Dateinamen. Dabei kann davon ausgegangen werden, dass die übergebenen Argumente folgende Bedingungen erfüllen:

1. Wird ein Dateiname angegeben, so existiert eine Datei mit diesem Namen. Andernfalls gibt es eine Datei mit dem Namen `labyrinth.dat`. Die jeweilige Datei wird im Folgenden als *Eingabedatei* bezeichnet und hat das oben beschriebene Format.
2. Für $p \in \{s,t\}$ steht an p_1 -ter Stelle in der p_0 -ten Zeile der Eingabedatei ein `P`.

Rückgabewert Die Anzahl der Schritte von (s_0, s_1) zu (t_0, t_1) in dem Labyrinth in der Eingabedatei, falls ein solcher Weg existiert; andernfalls soll `-1` zurückgegeben werden.

Beispielaufufe (für obiges Labyrinth)

```
>>> abstand((0,9), (2,2))
11
>>> abstand((0,9), (0,7))
-1
```