

پروژه پردازش زبان

توضیحات کد:

بخش initial assignments:

آرایه distArr: در این آرایه فاصله ویرایشی در میان دو رشته قرار میگیرد در ابتدا ستون اول و سطر اول برابر با شماره سطر و ستون متناظر مقداردهی می شود: (مثال برای کلمه مبدأ 5 و کلمه مقصد 4 حرفی)

```
[[0. 1. 2. 3. 4.]  
 [1. 0. 0. 0. 0.]  
 [2. 0. 0. 0. 0.]  
 [3. 0. 0. 0. 0.]  
 [4. 0. 0. 0. 0.]  
 [5. 0. 0. 0. 0.]
```

چرا که اگر رشته هدف دارای طول صفر باشد به تعداد طول مبدأ delete و اگر مبدأ دارای طول صفر باشد به تعداد طول مقصد insert بایستی انجام شود

message list: در این لیست تک تک تصمیمات مبنی بر حذف یا اضافه یا تعویض دو حرف قرار میگیرد این لیست پس از اجرای الگوریتم در مرحله back trace پرمی شود

decisionMadeArr: در این آرایه در هریک از مراحل هریک از تصمیمات یعنی «حذف» یا «اضافه» یا «تعویض» که گرفته شود درایه مربوطه یک می شود این آرایه در مرحله back trace پر می شود

پس از آن مقدار هزینه برای حذف و اضافه کردن یا جابه جایی دو حرف قرار داده شده است نامگذاری potentialSubsCost به این علت است که اگر دو حرف جابه جا شده یکسان باشند هزینه ای در بر نخواهد داشت و این هزینه تنها در حالتی اضافه می شود که دو حرف جابه جا شده یکسان نباشند.

بخش algorithm implementation:

در این بخش distArr مطابق الگوریتم و قوانین زیر پر می شود:

$$D[i, j] = \min \begin{cases} D[i-1, j] + \text{del-cost}(\text{source}[i]) \\ D[i, j-1] + \text{ins-cost}(\text{target}[j]) \\ D[i-1, j-1] + \text{sub-cost}(\text{source}[i], \text{target}[j]) \end{cases}$$

آرایه distArr پس از اجرای الگوریتم برای arash و arsh اینچنین خواهد بود:

```
[[0. 1. 2. 3. 4.]
 [1. 0. 1. 2. 3.]
 [2. 1. 0. 1. 2.]
 [3. 2. 1. 2. 3.]
 [4. 3. 2. 1. 2.]
 [5. 4. 3. 2. 1.]]
```

میبینیم که هزینه 1 شده است علت این است که کافیسیت یک a از arash حذف شود تا به مقصد برسیم.

بخش back trace:

در این بخش پس از بدست آمدن جواب در distArr از درایه انتهایی (یعنی سطر آخر ستون آخر) شروع و چک میکنیم چه تصمیمی گرفته شده است و به ازای آن پیام لازم را در message list ثبت و به درایه عقبتر باز میگردیم

درانتها آنچه در message list ثبت شده را نمایش میدهیم
همچنین decisionMadeArr نشان دهنده مسیر طی شده از مبدأ تا مقصد است.
خروجی:

```
converting arash to arsh
made a substitution between a and a
made a substitution between r and r
deleted the a
made a substitution between s and s
made a substitution between h and h
```

```
the path:
[[1. 0. 0. 0. 0.]
 [0. 1. 0. 0. 0.]
 [0. 0. 1. 0. 0.]
 [0. 0. 1. 0. 0.]
 [0. 0. 0. 1. 0.]
 [0. 0. 0. 0. 1.]]
```

همانطور که میبینیم در آرایه مسیر به جز یک قسمت کلاً حرکت مورب بوده یعنی تعویض انجام شده و چون حروف یکسان تعویض شده اند هزینه اش صفر بوده و تنها در یک قسمت مورب نبوده که علتش حذف a وسط از arash است.

باتشکر