

[Início](#)[Descrição](#)[Editor](#)[Submissões](#)

Fri May 3 18:01:43 2024 Rodrigo Mendes (up202308366)

[Logout](#)

## Exercício 2: Preencher um parágrafo

Escreva uma função

```
fillWords :: Int -> [String] -> [[String]]
```

que recebe uma lista de palavras (o segundo argumento) e preenche linhas sem exceder o comprimento máximo (o primeiro argumento). O resultado deve ser uma lista de linhas, em que cada linha é uma lista de palavras.

Pode assumir que nenhuma palavra tem comprimento superior ao máximo.

### Exemplo

```
ghci> let list = words "It was the best of times, it was the worst of times, it was the age of wisdom,
it was the age of foolishness"
ghci> fillWords 40 list
[["It","was","the","best","of","times","it","was","the"],
["worst","of","times","it","was","the","age","of"],
["wisdom","it","was","the","age","of","foolishness"]]
ghci> fillWords 70 list
[["It","was","the","best","of","times","it","was","the","worst","of","times","it","was","the","age","of"],
["wisdom","it","was","the","age","of","foolishness"]]
```

### Sugestão

Comece por definir uma função auxiliar

```
wordsThatFit :: Int -> [String] -> Int
```

que recebe um comprimento máximo e uma lista de palavras e determina quantas palavras cabem no comprimento dado.

Submissões: *after Wed Apr 17 00:00:00 2024 and before Mon May 6 00:00:00 2024*