

## Aula prática 6

---

Este desafio tem como objetivo aplicar conhecimentos adquiridos até ao momento na unidade curricular Programação 2. Pretende-se com este problema que os estudantes desenvolvam um programa capaz de realizar a tarefa solicitada no menor tempo possível, usando para isso as bibliotecas fornecidas no Moodle (vetor, lista ligada, fila e pilha).

---

### Desafio 3 - Panquecas

Dado uma torre de panquecas, desenvolva um programa que indica como é que a torre de panquecas deve ser ordenada de modo a que a maior panqueca esteja na base e a mais pequena no topo. O tamanho das panquecas (diâmetro) é classificado por uma letra do alfabeto, sendo A o diâmetro mais pequeno e Z o maior. Considere que na torre todas as panquecas têm tamanhos diferentes.

A ordenação da torre das panquecas é realizada através de uma sequência de *flips*. Um *flip* consiste em colocar uma espátula entre duas panquecas da torre e inverter as panquecas sobre a espátula (inversão da sub-torre). Um *flip* é especificado indicando a posição da panqueca da base da sub-torre. A panqueca no topo da torre completa tem a posição 1 e a da base tem a posição  $n$ .

Uma torre é especificada indicando-se o diâmetro de cada panqueca na torre pela ordem em que se encontram (do topo para a base).

Por exemplo, considere as três torres seguintes (H é a panqueca no topo na torre da esquerda, posição 1). A torre da esquerda pode ser transformada na torre do meio através da operação *flip*(4). A torre do meio pode ser transformada na da direita através de *flip* (6).

1	H	G	B
2	D	F	E
3	F	D	H
4	G	H	D
5	E	E	F
6	B	B	G

A entrada consiste numa sequência de 15 torres de panquecas, uma torre por linha e cada uma contendo até 26 panquecas. Para cada torre é indicado o diâmetro das panquecas (que varia entre A e Z), começando pelo topo e terminando na base.

A saída do seu programa deve apresentar a solução de ordenação para todas as torres de panquecas. A solução deve apresentar a torre original (numa linha, do topo para a base) seguida da sequência de *flips* necessários para ordenar a torre (apresente a operação de flip e também a torre resultado após cada *flip*).

*Exemplo*

Input	Output
W L R B M Q H C D A	Original pancake tower W L R B M Q H C D A flip(10): A D C H Q M B R L W flip(8): R B M Q H C D A L W flip(9): L A D C H Q M B R W flip(6): Q H C D A L M B R W flip(8): B M L A D C H Q R W flip(2): M B L A D C H Q R W flip(7): H C D A L B M Q R W flip(5): L A D C H B M Q R W flip(6): B H C D A L M Q R W flip(2): H B C D A L M Q R W flip(5): A D C B H L M Q R W flip(2): D A C B H L M Q R W flip(4): B C A D H L M Q R W flip(2): C B A D H L M Q R W flip(3): A B C D H L M Q R W Sorted pancake tower A B C D H L M Q R W

*Adaptado de UVa Online Judge*