

Indentador de BF

A solução utiliza de um procedimento executado em várias etapas para conseguir um tempo de execução $\mathcal{O}(n)$:

1. A entrada é analisada caractere a caractere. Caso o caractere atual seja um comando, empilhamos em um vetor de comandos. Caso não seja um comando e não seja um espaço, lemos até encontrar o fim da linha ou outro comando. Após a leitura, removemos quaisquer espaços no final e adicionamos essa segmento como comentário do último comando encontrado. O enunciado garante que o primeiro caractere é sempre um comando, então não temos que se preocupar com a sua ausência. Também indicamos em um vetor que o próximo comando possui um comentário anterior, incrementando a posição.
2. Fazemos a soma de prefixos no vetor que indica se os comandos tem comentários anteriores ou não. Na prática, isso significa que cada posição do vetor agora possui a informação de quantos comentários existem do começo até o comando.
3. Agora realizamos a identificação do nível de indentação de cada comando. Usando uma pilha, empilhamos todos os '[' que encontrarmos. Ao encontrar um ']', pegamos a posição do último '[' da pilha, desempilhamos, e verificamos se a quantidade de comentários cresceu entre o o comando '[' e o comando ']'. Se sim, aplicamos o procedimento de criar um nível indentação naquela posição, e remover um nível de indentação no final, além de aplicarmos o isolamento dos comandos em sua própria linha.
4. Fazemos a soma de prefixos no vetor de indentação para conseguirmos o nível real de indentação, visto que no passo anterior apenas somamos no começo e retiramos no final.
5. Resta agora só imprimir cada comando com a sua indentação, comando e comentário.