

Imagens

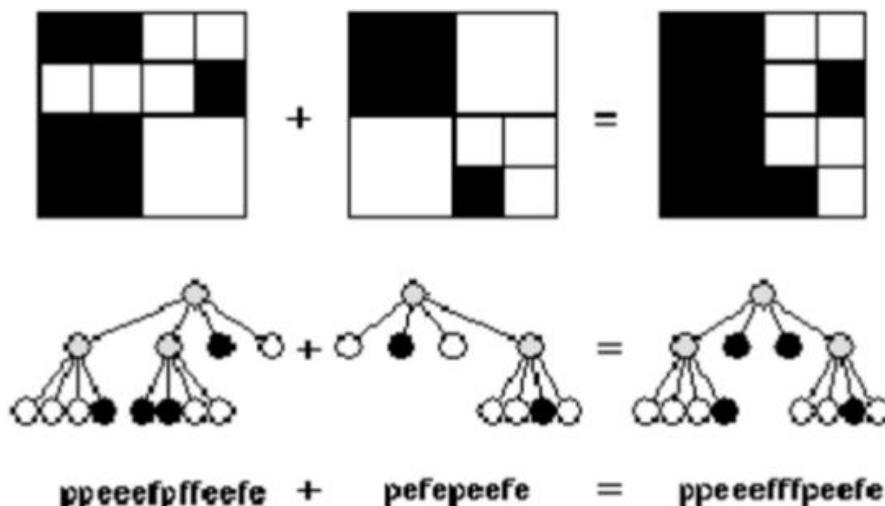
Descrição

Uma árvore quaternária é um formato de representação muito usado para codificar imagens. A ideia fundamental por trás da utilização de uma árvore quaternária para esse propósito é que qualquer imagem pode ser dividida em 4 quadrantes. Por sua vez, cada quadrante pode também ser dividido em 4 sub-quadrantes e assim sucessivamente. Em uma árvore quaternária a imagem é representada por um nó pai, enquanto que os quadrantes da imagem são representados por 4 nós filhos em uma ordem predeterminada (nordeste, noroeste, sudoeste, sudeste).

Nas horas vagas, Pedro é um artista que usa seu computador para produzir obras de arte modernas. Pedro gosta de trabalhar com imagens em preto e branco de 1024 pixels (grade de 32 X 32). Uma das técnicas que Pedro usa é sobrepor duas imagens para formar uma nova imagem. Na imagem resultante, um pixel é preto se ele era preto em, ao menos, uma das imagens componentes, caso contrário, ele é branco. Ou seja, preto + preto = preto; preto + branco = preto e branco + branco = branco. Pedro acredita que para resultados de qualidade o que interessa é a quantidade de pixels pretos contidos na imagem gerada. Portanto, ele gostaria muito de saber previamente a quantidade de pixels pretos que uma imagem resultante teria a partir da sobreposição de duas outras imagens quaisquer (sempre de tamanho 32 X 32 pixels).

Ajude Pedro a criar obras de arte moderna de qualidade. Crie um programa que recebe 2 representações de árvores quaternárias (por meio de strings), simbolizando 2 imagens, e calcule o número de pixels pretos na imagem resultante considerando a sobreposição dessas 2 imagens representadas pelas árvores quaternárias recebidas como entrada.

Na Figura abaixo, é possível observar um exemplo de sobreposição de duas árvores quaternárias de ordem 4 X 4. As strings representando cada árvore aparecem na parte de baixo da figura.



ATENÇÃO: É de uso obrigatório a utilização da estrutura de árvore quaternária para a resolução do problema.

Entrada

A primeira linha da entrada é um inteiro positivo N, que especifica a quantidade de casos de teste a serem tratados. A segunda e terceira linhas são strings representando árvores quaternárias (em pré-ordem). Nas strings, a letra "p" indica um nó pai, a letra "f" indica um pixel preto (*full*) e a letra "e" indica um pixel branco (*empty*). Cada string é considerada uma árvore quaternária válida. A profundidade da árvore é no máximo 5.

Exemplo de entrada:

```
2 (a quantidade de casos de teste)
ppeeefpffee (string representando a primeira árvore do caso 1)
pefepee (string representando a segunda a árvore do caso 1)
ppeeefpffee (string representando a primeira árvore do caso 2)
pefepee (string representando a segunda a árvore do caso 2)
```

Saída

Para cada caso de teste, imprima em uma linha o texto "P X pixels pretos.", onde P é o caso de teste e X é a quantidade de pixels pretos encontrados na imagem resultante.

Exemplo de saída:

```
1 640 pixels pretos.
2 864 pixels pretos.
```

Exemplos de Entrada e Saída

Entrada 1

```
1
peeef
peeef
```

Saída 1

```
1 512 pixels pretos.
```

Entrada 2

```
2
peeef
peepefe
peeef
peeef
```

Saída 2

```
1 384 pixels pretos.
2 512 pixels pretos.
```