

# Baralho

Nome do arquivo: “**baralho.x**”, onde **x** deve ser **c**, **cpp**, **pas**, **java**, **js** ou **py**

Uma gráfica iniciou a produção de cartas de baralho. Cada baralho produzido deve ser um *baralho completo*, ou seja, deve ter exatamente 52 cartas, compreendendo quatro naipes (Copas, Espadas, Ouros e Paus), com treze cartas em cada naipe (Ás, 2, 3, 4, 5, 6, 7, 8, 9, 10, Valete, Dama e Rei).

Um robô coleta cartas produzidas pelas máquinas impressoras e cortadoras e as agrupa em conjuntos de 52 cartas, preparando o baralho para ser embalado para venda. A empresa deseja garantir que cada baralho embalado seja um *baralho completo* e precisa de sua ajuda.

Dada a lista das cartas de um baralho pronto para ser embalado, escreva um programa para verificar se há cartas faltando ou duplicadas no baralho.

## Entrada

A primeira linha da entrada contém uma cadeia de caracteres que descreve as cartas do baralho. Cada carta é descrita usando três caracteres, no formato **ddN** onde **dd** são dois dígitos decimais (de 01, representando a carta Ás, a 13, representando a carta Rei) e **N** é um caractere entre C, E, U e P, representando respectivamente os naipes Copas, Espadas, Ouros e Paus). Note que o caractere que representa o naipe Ouros é U (e não O), para não confundir com o dígito zero.

## Saída

Seu programa deve produzir exatamente quatro linhas na saída, cada linha correspondendo aos naipes Copas, Espadas, Ouros, e Paus, nessa ordem. Para cada naipe, se o conjunto de cartas está completo (ou seja, se exatamente 13 cartas com valores de 01, 02, 03, ..., 12, 13 estão presentes), seu programa deve produzir o valor 0; se o conjunto de cartas tem alguma carta duplicada, seu programa deve produzir a palavra **erro**; se o conjunto de cartas tem cartas faltando, seu programa deve imprimir o número de cartas que faltam.

## Restrições

- $3 \leq$  comprimento da cadeia de caracteres na entrada  $\leq 156$
- para toda carta **ddN**,  $01 \leq dd \leq 13$  e **N** é C, E, U ou P.

## Informações sobre a pontuação

- Para um conjunto de casos de teste valendo 20 pontos, não há cartas duplicadas, há apenas cartas faltando.

## Exemplos

<b>Exemplo de entrada 1</b>  11P01C02C01U02U03U04U	<b>Exemplo de saída 1</b>  11 13 9 12
<b>Exemplo de entrada 2</b>  13P02P01P03P04P05P06P07P08P09P10P11P12P	<b>Exemplo de saída 2</b>  13 13 13 0

Exemplo de entrada 3	Exemplo de saída 3
01C02C03C04C05C07C09C10C11C02E02E03E11U	4 erro 12 13