Ordenação

Oficina de AEDs2 - Prof. Matheus Pereira

1. Exercício em Java: Ranking dos Melhores Jogadores do Campeonato Brasileiro

Ao final do campeonato brasileiro, deseja-se criar um ranking dos melhores jogadores baseado no número médio de participações em gol por partida (gols + assistências). O sistema deve processar um arquivo de entrada e gerar um ranking ordenado.

Implemente um programa em Java que leia informações sobre jogadores e gere um ranking dos melhores jogadores seguindo estes critérios:

1. **Cálculo da Pontuação**:
   - Para cada jogador, calcule a média de participações em gol por partida.
   - Participação em gol = gols + assistências.
   - Média = total de participações / número de partidas.
2. **Critério de Ordenamento**:
   - Ordene os jogadores da maior média para a menor.
   - Em caso de empate na média, o jogador com **mais gols** totais fica à frente.
   - Se persistir o empate, ordene pelo nome em ordem alfabética.
3. **Entrada**:
   - Primeira linha: número inteiro com a quantidade de jogadores
   - Linhas seguintes: dados dos jogadores no formato:

     `NomeDoJogador gols1 assist1 gols2 assist2 ... golsN assistN`

     Onde golsK e assistK indicam, respectivamente, a quantidade de gols e assistências do jogador na partida K.
   - Última linha: FIM.
4. **Saída**:
   - Imprima a string "Ranking de jogadores:" e, em sequência, exiba o ranking ordenado com o formato:

     `Posição. NomeDoJogador - Média: X - Gols: Y`

- A média deve ser impressa com 2 caracteres decimais. Use o seguinte comando para formatar o número de casas decimais da variável numérica *media*:

```
String formatada = String.format("%.2f", media);
```

Exemplo de Entrada

```
10
KaioJorge 2 1 1 0 3 2 0 1
VitorRoque 1 2 0 1 2 0 1 1 0 2
Pedro 0 1 1 0 2 1 1 1
Neymar 1 0 0 1 0 1
GustavoScarpa 1 1 0 2 1 1 0 1
Cano 1 0 2 1 0 0 1 1
Hulk 0 2 1 1 1 0 2 1
MatheusPereira 3 2 2 1 1 3 2 2 0 1
YuriAlberto 0 0 1 0 0 1 0 0
Vegetti 1 1 0 2 1 0 2 1
FIM
```

Exemplo de Saída

```
Ranking de jogadores:
1. MatheusPereira - Media: 3.40 - Gols: 8
2. KaioJorge - Media: 2.50 - Gols: 6
3. Hulk - Media: 2.00 - Gols: 4
4. Vegetti - Media: 2.00 - Gols: 4
5. VitorRoque - Media: 2.00 - Gols: 4
6. Pedro - Media: 1.75 - Gols: 4
7. GustavoScarpa - Media: 1.75 - Gols: 2
8. Cano - Media: 1.50 - Gols: 4
9. Neymar - Media: 1.00 - Gols: 1
10. YuriAlberto - Media: 0.50 - Gols: 1
```

2. Exercício em Java ou C: Lista telefônica econômica

Adaptado de `https://judge.beecrowd.com/en/problems/view/1211`

Devido ao grande número de reclamações, a companhia telefônica de São Petersburgo é forçada a investir pesadamente na melhoria de seus serviços. Para isso, a empresa decidiu reduzir o orçamento de alguns setores para aumentar outros mais essenciais. Um dos setores que terá seu orçamento reduzido é a impressão de listas telefônicas.

Com um orçamento reduzido, o setor de impressão de listas telefônicas não pode comprar toner suficiente para imprimir as listas completas. Como os números de telefone são impressos alinhados verticalmente, foi sugerida a seguinte solução: a partir do segundo número de telefone impresso, os dígitos iniciais do próximo número a ser impresso que coincidirem com o número acima são omitidos, deixando apenas um espaço em branco.

Por exemplo, para os números 535456, 535488, 536566 e 835456, a impressão seria feita da seguinte forma:

```
5 3 5 4 5 6
        8 8
    6 5 6 6
8 3 5 4 5 6
```

Note que esta impressão economizou a impressão de 6 caracteres. A companhia telefônica também considerou não imprimir os sufixos repetidos, mas em testes percebeu que a resposta não era boa para o usuário e, portanto, decidiu fazer apenas a eliminação de prefixos. Para determinar se a economia será suficiente, a indústria de impressão quer saber o número máximo de caracteres que podem ser omitidos. No entanto, como em qualquer grande cidade, há vários números de telefone e eles não querem gastar horas de trabalho para calcular esse valor manualmente. Então cabe a você, novo funcionário da empresa, automatizar o cálculo da economia de toner, ou seja, o número de caracteres economizados.

Entrada

A entrada consiste em vários casos de teste e termina com o final de arquivo (EOF). Cada caso de teste contém um inteiro N, que informa o número de telefones na lista. As próximas N ($1 \leq N \leq 10^5$) linhas têm, cada uma, um número de telefone X_i, com até 200 caracteres. Em um mesmo caso de teste, os números de telefone têm a mesma quantidade de caracteres. Um número de telefone pode começar com o caractere '0'.

Saída

Para cada caso de teste, imprima uma linha indicando o número máximo de caracteres economizados por esse processo.

Exemplo de Entrada

```
3
12345
12354
54321
```

```
4
536566
535488
535456
835456
```

Exemplo de saída

```
3
6
```

3. Exercício em C

Imagine que você está desenvolvendo um sistema simples de **cadastro de participantes para uma corrida de rua**. A organização do evento deseja gerar uma lista de largada ordenada com os seguintes critérios:

1. Todas as **mulheres** devem aparecer antes dos homens.
2. Dentro de cada grupo (mulheres e homens), a lista deve ser ordenada alfabeticamente pelo **sobrenome**.
3. Caso dois participantes possuam o mesmo sobrenome, o critério de desempate será o **primeiro nome**, também em ordem alfabética.

Especificações do Programa

Implemente em linguagem C um programa que:

1. Leia a quantidade de participantes (n, onde $1 \leq n \leq 100$).
2. Para cada participante, leia:
   - o **primeiro nome** (máximo de 20 caracteres),
   - o **sobrenome** (máximo de 30 caracteres),
   - um **caractere indicando o sexo** (`F` para feminino, `M` para masculino).
3. Ordene os participantes utilizando um método de ordenação à sua escolha, implementado manualmente (não é permitido usar funções prontas de ordenação).
4. Imprime o númeo de participantes e exibe a lista final, respeitando os critérios estabelecidos.

Exemplo de Entrada

```
6
Maria Silva F
Marcelo Souza M
Ana Silva F
Pedro Oliveira M
Lucas Souza M
Beatriz Almeida F
```

Exemplo de Saída

```
6
Beatriz Almeida
Ana Silva
Maria Silva
Pedro Oliveira
Lucas Souza
Marcelo Souza
```