

1ª Avaliação - Servlets / JSP / JSF

- 1) **(3,0 pontos)** Escreva um servlet para calcular as permutação e as combinações de K elementos a partir de um conjunto com N elementos, com os dados informados pelo usuário no formulário HTML mostrado abaixo. O servlet deve calcular os valores e colocá-los em variáveis de seção. Deverá ser gerada uma mensagem de erro na seção se o usuário informar valores incorretos (vazio ou não numéricos) para algum dos valores. Deve haver o redirecionamento para uma página JSP que mostre os resultados calculados. Em caso de dados incorretos, deve voltar para a página inicial, com a(s) mensagem(ns) de erro, permitindo ao usuário que informe os dados novamente.

```
<form action="calcular" method="POST">
  Tamanho do conjunto (N): <input type="text" name="N" size="3"><br/>
  Elementos a combinar(K): <input type="text" name="K" size="3"><br/>
  <input type="submit" value="Enviar"/>
</form>
```

Fórmulas:

$$A_{K,N} = \frac{N!}{(N-K)!} \qquad C_{K,N} = \frac{N!}{K!(N-K)!}$$

Para calcular o fatorial, use a função:

```
public static long fatorial( int x ) {
    long fat = 1;
    for (int i = x; i > 1; i--) { fat = fat * i; }
    return fat;
}
```

- 2) **(7,0 pontos)** Utilizando classes Java para representar e armazenar os dados, Beans JSF o processamento das requisições e JSF para a apresentação e interação com o usuário, faça uma aplicação web para o jogo pedra-papel-tesoura (Jo-Ken-Po). A dinâmica deve ser a seguinte:

a) Ao acessar a primeira tela, o usuário deverá identificar-se. Numa classe estática, todos os jogadores devem ficar registrados, com seu desempenho (número de jogos, número de vitórias, número de empates e de derrotas). Ao digitar o nome e entrar no jogo, o nome deve ser buscado na lista de jogadores. Se não existir deve ser criado um novo jogador.

b) Na tela de jogo, o usuário escolhe uma das três opções, e clica em "Jogar!". Um servlet deverá então sortear uma das opções de jogo aleatoriamente. Use

```
Random rnd = new Random();
int ale = rnd.nextInt(3) + 1;
```

para gerar um número de 1 a 3. Deve então mostrar as escolhas dos jogadores (usuário e computador) e escrever o resultado: "Você venceu.", "Computador venceu." ou "Empate". Devem ser mostradas estatísticas do jogo: número de partidas, número de vitórias para cada jogador e número de empates.

c) Crie um botão para ver as estatísticas de todos os jogadores, na forma de uma tabela. Na tela de jogo e de estatísticas, crie um botão para que o usuário apague suas estatísticas.




Sugestões de telas:

Bem-vindo. Identifique-se para criar/recuperar jogador.

Nome:

Pedra - Papel - Tesoura



Escolha seu jogo:

☐  ☒  ☐ 

Pedra - Papel - Tesoura

Resultado

Sua Jogada Computador

 X 

Computador venceu

Estatísticas:

Jogos: 13 Suas vitórias: 4 Vitórias do Computador: 5 Empates: 4

Estatísticas dos Jogadores

Nome	Vitórias	Empates	Derrotas
Ana	10	0	1
Juca	18	5	0
Zeca	23	12	10
Pafuncio	0	1	7

Obs: as imagens estão no arquivo imgs.zip. Nas opções, basta usar o nome da imagem. Para a jogada, pode criar um método que retorne o nome da imagem das opções do jogador e do computador. Ex:

`<h:graphicImage value="imgs/${controle.jogadaComputador}.png"/>` e
`<h:graphicImage value="imgs/${controle.jogadaJogador}.png"/>`