

Licenciatura em Engenharia Informática e de Telecomunicações
Departamento de Tecnologias de Informação e Comunicação
Programação I

Semana 6 – Aulas práticas 11 e 12

Turmas: I12, 14

Data: - 22 de Setembro 2020

Docentes: Grupo de Disciplina

PARTE A

1. Explique em que momento deve utilizar o *casting*?
2. Considerando a string palavra valendo “autodeterminação”, quais serão os resultados das expressões `palavra.substring(11)`, `palavra.substring(6,13)` e `palavra.substring(4,9)`?

PARTE B

Parte Prática Casting

3. Caso seja necessário realizar o casting, altere as instruções:

- a) `double d = 3.14;`
`int i = d;`
- b) `long x = 10000;`
`int i = x;`
- c) `int i = 5;`
`double d = i;`
- d) `double d = 5;`
`float f = 3;`
`float x = f + d;`

Class Math e Random

4. Crie uma classe que simule a jogada de um dado (de seis lados) dez vezes e mostre o resultado na tela.
5. Crie uma classe que calcule quantos metros cúbicos de água suporta uma determinada caixa-d'água em forma de cubo (todos os lados são iguais). O usuário deverá informar o valor do lado e o volume será calculado pela fórmula: **$V = l^3$** .

6. Crie uma classe para determinar o Área de uma circunferência.
7. Crie uma classe para determinar o volume de um Cone.
8. Crie uma classe para determinar o volume de um Cilindro.
9. Crie uma classe para calcular a raiz quadrada dos números 900 e 30.25.
10. Crie uma classe para calcular a potência de 5.5 elevado a 2 e 25 elevado a 0.5.
11. Crie uma classe que gere 5 cartões de loteria com seis números em cada um.
12. Crie uma classe para fazer o arredondamento dos seguintes valores: 5.2, 5.6 e -5.8 para o valor inteiro mais próximo.
13. Crie uma classe para fazer o arredondamento dos seguintes valores: 5.2, 5.6 e -5.8 para o valor inteiro anterior.
14. Faça um programa que receba um número aleatório de zero até um numero digitado pelo usuário. O programa deverá exibir o seno, cosseno, tangente, raiz quadrada e potência do número gerando pelo método *Math.random()*.
15. Fazer uma classe Sorteio para:
 - Sortear um número de 0 a 1000 (dica: usar *Math.random()*)
 - Pedir um palpite ao usuário. Se ele errar, informar se o palpite é maior ou menor do que o número sorteado.
 - Pedir novos palpites até que o usuário acerte e, depois disso, mostrar em quantas tentativas ele acertou.

Classe String

16. Faça um programa que permita ao usuário digitar o seu nome e em seguida imprima o nome do usuário de trás para frente utilizando somente letras maiúsculas. Dica: lembre-se que ao informar o nome o usuário pode digitar letras maiúsculas ou minúsculas.
17. Dada uma *string* com uma frase informada pelo usuário (incluindo espaços em branco), conte:

- quantos espaços em branco existem na frase.
- quantas vezes aparecem as vogais a, e, i, o, u.

18. Desenvolva um jogo em que o usuário tenha que adivinhar uma palavra que será mostrada com as letras embaralhadas. O programa terá uma lista de palavras lidas de um arquivo texto ou de um vector e será escolhida uma aleatoriamente. O jogador terá seis tentativas para adivinhar a palavra. Ao final a palavra deve ser mostrada na tela, informando se o usuário ganhou ou perdeu o jogo.

19. Escreva um programa que dado um valor numérico digitado pelo usuário (armazenado em uma variável inteira), imprima cada um dos seus dígitos por extenso. Exemplo:

```
Entre o número: 4571
Resultado: quatro, cinco, sete, um
```

20. Escreva uma classe em Java que represente o nome de uma Instituição, composto de três strings (nome próprio, nome do meio e nome da família). Deve no fim imprimir somente as iniciais, imprimir também as iniciais do nome próprio e do meio e o nome de família completo. Por exemplo, se o nome da pessoa representado por essa classe for “ Instituto de Transportes e Comunicação”, deve ser apresentado “itc” e “I.T.Comunicação”. Para facilitar, considere armazenar os três nomes em strings separadas.

21. Escreva na classe que reverta a ordem dos caracteres de uma string Um exemplo: se a string “Java” for passada , ele deve retornar a string “avaJ”. Dica: Use um laço for ou while e o método charAt, e crie uma nova string que receberá os caracteres na ordem invertida. Não use mecanismos da classe StringBuffer.

22. Escreva uma classe que receba uma string e um carácter como argumento, e imprima todas as substrings da string passada que terminem com o carácter passado. Por exemplo, se a string “indeterminadamente” e o carácter “e” forem passados como argumentos, o método deverá imprimir as strings “inde”, “indete”, “indeterminadame”, “indeterminadamente”.

23. Escreva um programa que, a partir de um nome informado pelo usuário, exiba suas iniciais. As iniciais são formadas pela primeira letra de cada nome, sendo que todas deverão aparecer em maiúsculas na saída do programa. Note que os conectores e, do, da, dos, das, de, di, du não são considerados nomes e, portanto, não devem ser considerados para a obtenção das iniciais. As iniciais devem ser impressas em maiúsculas, ainda que o nome seja entrado todo em minúsculas. Exemplos:

Maria das Graças Pimenta => MGP

João Carlos dos Santos => JCS

24. Em um jogo de tabuleiro chamado PalavrasCruzadas, cada palavra formada por um jogador vale um certo número de pontos, que depende das letras usadas. O número de pontos para as letras do alfabeto é dado por:

- Para cada letra „Q” ou “Z” na palavra some 10 pontos
- Para cada letra “J” ou “X” na palavra some 8 pontos.
- Para cada letra “K” na palavra some 5 pontos.
- Para cada letra “F”, “H”, “V”, “W” ou “Y” na palavra some 4 pontos.
- Para cada letra “B”, “C”, “M” ou “P” na palavra some 3 pontos.
- Para cada letra “D” ou “G” na palavra some 2 pontos.
- Para todas as outras letras some 1 ponto.

Caracteres que não sejam letras devem ser ignorados. Caracteres minúsculos devem ser convertidos para maiúsculos, e caracteres acentuados devem ser desacentuados. Por exemplo, o número de pontos para a palavra “Java” no jogo será $8+1+4+1=14$ pontos. Escreva uma classe PalavrasCruzadas em Java que receba uma *string* e informe o número de pontos que esta *string* valeria no jogo.