

USJT – 2016/2 - Programação de Computadores - Laboratório de Exercícios

Professores: Celso, Hamilton e Vaz

**Aula: 01**

Assunto: Estrutura básica de um programa Java, Entrada e Saída, Variáveis, Atribuição, Conversão de Strings

**Critérios de Avaliação do Laboratório**

Entrega de exercícios: 1 ponto por aula

Prova prática final: 1 ponto

A nota final do laboratório será a soma da nota da prova mais a média das entregas de exercícios, perfazendo um total de, no máximo, **2 pontos**.

**Dinâmica do Laboratório**

Os alunos recebem o roteiro de exercícios do laboratório e trabalham individualmente na resolução dos problemas propostos. Ao final da aula, devem mostrar ao professor os exercícios resolvidos e irão pontuar de acordo com a meta atingida: 0,0 (zero) ponto, caso não tenham resolvido nenhum exercício; 0,5 (meio) ponto caso tenham resolvido os exercícios que valem 0,5 ponto; e 1,0 (um) ponto caso resolvam todos os exercícios propostos para a aula, com exceção dos complementares, que não pontuam.

Obs.: O professor deve considerar a solução apresentada pelo aluno para pontuá-lo, mesmo que esteja incorreta, desde que esta demonstre a intenção do aluno de resolver licitamente o problema.

**Papel do Professor**

Ao iniciar a aula o professor deve mostrar e comentar com os alunos, linha a linha, o código do exercício resolvido. Na sequência, comentar os enunciados com os alunos, detendo-se naqueles que apresentarem alguma dificuldade de entendimento sobre o que fazer por parte deles.

Além disso, o professor deve tempo todo percorrer o laboratório avaliando as soluções, tirando dúvidas dos que pedem e oferecendo ajuda para os alunos. Caso perceba que a dúvida é comum, pode resolvê-la na lousa ou no datashow.

Entretanto, o professor deve procurar evitar, a todo custo, consumir muito tempo da aula explicando o conteúdo para os alunos novamente, como se estivesse dando a

aula teórica novamente. Este laboratório é para os alunos praticarem sozinhos e tirarem as dúvidas que forem surgindo durante a prática.

### **Problemas Propostos:**

- todos os problemas podem ser resolvidos diretamente no método main.
- faça uma classe para cada solução.
- a nome da classe pode ser Solucao01a, Solucao1b, e assim por diante.
- não use Scanner para ler dados.
- os alunos podem consultar qualquer material (menos o Google).

### **Exercícios iniciais: valor 0,5 ponto**

Resolva os exercícios desta seção para conquistar 0,5 ponto

- 1) Crie um algoritmo que leia três palavras, separadamente, e as apresente na tela na ordem contrária em que foram digitadas, separadas por espaço.
- 2) Crie um algoritmo que leia um número inteiro, armazene-o em uma variável do tipo int e depois escreva na tela: "O valor do número digitado é: " e o valor do número.

### **Exercícios intermediários: valor 0,5 ponto**

Resolva os exercícios desta seção para conquistar mais 0,5 ponto

- 3) Crie um algoritmo que leia um número inteiro e armazene-o em uma variável do tipo int. Depois escreva na tela: "O valor do número digitado é: " e o valor do número.
- 4) Crie um algoritmo que leia 3 números que representem uma data, respectivamente dia, mês e ano. Depois imprima a data usando o formato dia/mês/ano.
- 5) Solicite ao usuário que digite seu peso no formato real. Depois imprima "O peso informado foi " + peso + " kg."

### **Exercícios complementares (para praticar depois da aula)**

Resolva os exercícios desta seção para aprimorar seus conhecimentos

- 6) Ler dez caracteres e imprimir a palavra formada por eles.
- 7) Leia 4 números inteiros, representando milhar, centena, dezena e unidade. Use o tipo caractere para garantir que não seja digitado mais de um número por vez. Depois concatene estes caracteres armazenando-os em uma String (lembre-se de que o jeito mais fácil de transformar algo em String é concatenando-o com uma String vazia ""). Imprima o resultado.

**Bibliografia**

LOPES, ANITA. GARCIA, GUTO. Introdução à Programação: 500 algoritmos resolvidos. Rio de Janeiro: Elsevier, 2002.

DEITEL, P. DEITEL, H. Java: como programar. 8 Ed. São Paulo: Prentice – Hall (Pearson), 2010.