

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

Professores: Rodrigo e Vaz

**Aula: 04**

Assunto: Laços de Repetição while; do – while; for

**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, Solucao01b, e assim por diante.
- não use Scanner para ler dados.
- não se preocupe em explicar o import, o static, o String[]; isso será feito adiante.
- os alunos podem consultar qualquer material.

**Kata**

- 1) Imprimir na tela os quadrados dos números inteiros de 1 a 20.
- 2) Entrar com números e imprimir o triplo de cada número. O algoritmo acaba quando entrar o número -999.

**Randori**

- 3) Entrar com inteiros enquanto forem maiores que zero e imprimir quantos números foram digitados.
- 4) Faça um algoritmo que mostre na tela a tabuada de um número qualquer digitado pelo usuário.
- 5) Dado um país A, com 5 milhões de habitantes e taxa de natalidade de 3% ao ano, e um país B, com 7 milhões de habitantes e taxa de natalidade de 2% ano, calcule e imprima o tempo necessário, em anos, para que a população do país A ultrapasse a do país B.

**Dojo - Conceitos básicos**

**Definição:** a palavra Dojo (pronuncia-se Dojô), em japonês, significa "local de treinamento". O termo, amplamente utilizado em artes marciais como o Kendo e o

Karatê, vem também sendo usado na prática de programação voltada à aprendizagem, chamado de *Coding Dojo*.

Existem alguns formatos do Dojo, como o *Kata* e o *Kake*. No nosso curso, usaremos o *Randori* e o *Kata*, os quais explico abaixo.

### **Randori**

- 1) Todos os alunos participam. É proposto um ou mais problemas para serem resolvidos e a programação é realizada por pares em apenas uma máquina (as outras ficam preferencialmente desligadas), projetando a tela para que todos vejam.
- 2) A solução deve ser gradual, obtida passo a passo, sendo explicada pelo par que está programando e deve ser entendida por todos. Nos métodos ágeis, isto é conhecido por *baby steps*.
- 3) A pessoa que está no teclado é o piloto e, seu par, o co-piloto. A cada cinco minutos o piloto volta para a plateia e o co-piloto assume a condição de piloto. Uma pessoa da plateia passa a assumir a posição de co-piloto. Caso a plateia seja muito grande, ambos podem ser substituídos por outra dupla.
- 4) O professor é o *sensei*. Seu papel é o de moderar o encontro, propor o problema, ajudar a dupla e a plateia encontrarem a solução, manter a ordem e a dinâmica do Dojo e avaliar a participação dos alunos.
- 5) A plateia deve prestar atenção atentamente na solução e se manter em silêncio. A plateia pode ajudar na solução caso a dupla que está no teclado esteja com dificuldades, mas isso deve ocorrer sempre mediante o estímulo do professor.
- 6) Ao final o professor deve fazer uma retrospectiva do Dojo, comentando as soluções obtidas, complementando o que foi ensinado e dando um feedback para a turma sobre a participação dela no Dojo.

O objetivo é fazer com que todos escrevam código, peguem gosto pela programação e aprendam a trabalhar coletivamente. O aluno ficará exposto e não poderá se omitir nem guardar as dúvidas para ele.

### **Kata**

- 1) O professor (*sensei*) inicia o Dojo apresentando a solução de um problema para os alunos, resolvendo-o no JGrasp e projetando o código na tela.
- 2) Os alunos, com os computadores ligados, vão copiando o código e entendendo a solução proposta.

O objetivo é preparar o aluno para a sessão de *Randori* que virá em seguida.

### Dojo - Regras:

**NOTA: APESAR DAS REGRAS, O DOJO DEVE SER UM AMBIENTE DESCONTRAÍDO E ESTIMULANTE, NO QUAL O ALUNO SE SINTA IMPELIDO A PARTICIPAR NÃO PELOS PONTOS, MAS SIM PELO GOSTO DE PROGRAMAR E DE ENCONTRA UMA SOLUÇÃO DE FORMA COLABORATIVA.**

1. Todas as máquinas do computador ficam preferencialmente desligadas, com exceção do computador que está sendo usado pelo piloto.
2. Todos os alunos da turma, se possível, devem se revezar no teclado a cada sessão de Dojo. Os alunos que não teclarem em uma aula tem preferência na seguinte.
3. O professor deve dar uma nota de 0,0 a 2,0 para cada aluno, em incrementos de 0,5, em cada sessão de Dojo. A nota final do laboratório será a média de todos os Dojos.
4. Todo aluno entra na sala com 2,0 pontos, que serão descontados segundo o seguinte critério de avaliação:
  - **pontualidade:** tolerância máxima de 20 minutos de atraso (penalidade de 0,5 ponto); mais de 40 minutos de atraso, não entra no laboratório (penalidade de 2,0 pontos).
  - **atenção quando na plateia:** estar atento, ajudar quando solicitado, ficar em silêncio, não usar o celular, não ligar o computador, não conversar com o colega. (penalidade de 0.5 ponto).
  - **desempenho como co-piloto:** ajudar o piloto na construção da solução (penalidade de 0,5 ponto).
  - **desempenho como piloto:** construir efetivamente a solução; a tolerância do professor com alunos que não tenham ideia do que fazer deve ser mais alta no começo e ir diminuindo conforme as aulas vão avançando (penalidade de 0,5 ponto).
5. Fica a critério do professor excluir um aluno do Dojo por mau comportamento a qualquer momento e retirar dele todos os pontos daquele Dojo.

### 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.

