



Trabalho – 01

Solução de um problema usando Threads

Rômulo Ferreira Douro

Programação Orientada a Objetos 2



SUMÁRIO

1	DESCRIÇÃO	3
1.1	RESULTADO ESPERADO	4
1.1.1	Estado inicial.....	4
1.1.2	Iniciar	5
1.1.3	Zerar.....	5
1.1.4	Atenção.....	6
2	ENTREGA DO TRABALHO	6
2.1	DICAS	7
3	AVALIAÇÃO DO TRABALHO	7
3.1	ATENÇÃO.....	7



1 DESCRIÇÃO

Uma empresa da área de TI contratou você e seu grupo de programadores para implementar a seguinte solução: **implementar um cronômetro digital com parada programada.**

Para tanto, a empresa forneceu os moldes de uso de seu cronômetro.

A tela de execução deve ser como na figura abaixo:

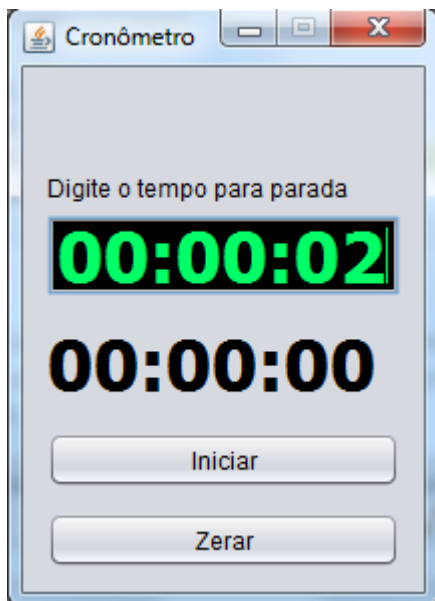
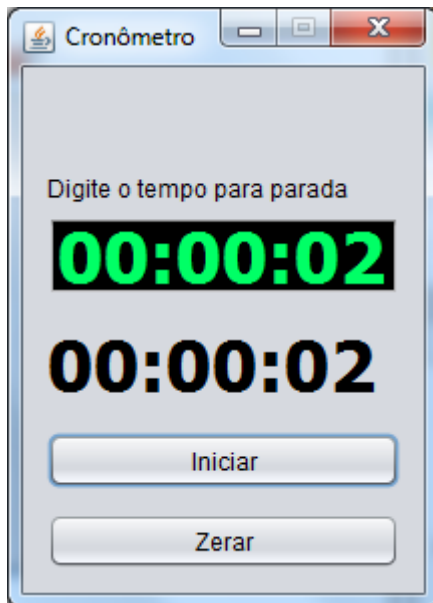


Figura 1 - Tela do cronômetro atualmente

Deseja-se que a string “00:00:00” seja substituída pelo tempo decorrido calculado e que seja atualizada a cada 1 segundo.

Ainda tem-se que, quando o tempo decorrido for igual ao tempo digitado em VERDE, o cronômetro será pausado indicando assim o término da contagem.



1.1 RESULTADO ESPERADO

O resultado esperado deve ser semelhante ao apresentado na imagem abaixo:

1.1.1 Estado inicial

Ao abrir o programa a tela deve ser mostrada com o cronômetro zerado e as opções de “Iniciar” e “Zerar” com a opção de digitação de tempo no espaço em verde.

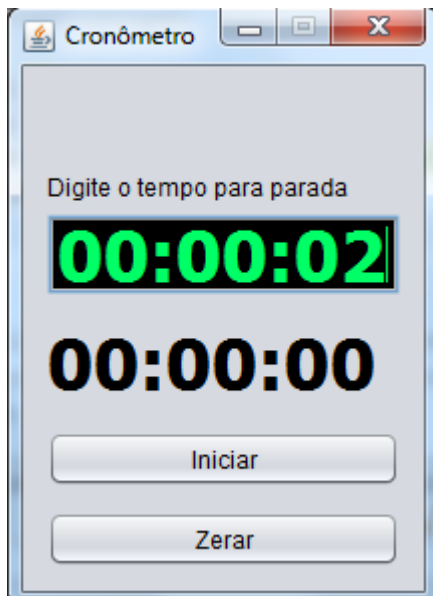


Figura 2 - Tela inicial da abertura do programa



1.1.2 Iniciar

A opção de “Iniciar” efetivamente inicia a contagem de tempo e atualiza o tempo decorrido na tela

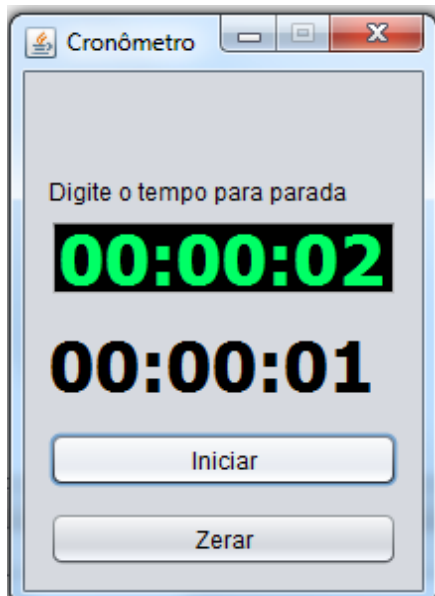


Figura 3 - Ao clicar no botão Iniciar

1.1.3 Zerar

Com o cronômetro em operação, ao selecionar a opção “Zerar” o mesmo deverá voltar ao seu estado inicial com o tempo zerado e o cronômetro “inativo”.

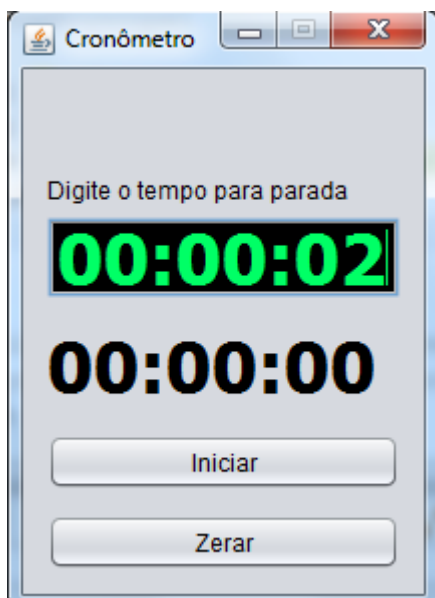


Figura 4 - Cronômetro zerado após clicar em "Zerar"



1.1.4 Atenção

O grupo deve ficar atento para que nenhuma atividade interfira no funcionamento da tela do programa fazendo com que a mesma fique “travada” ou “inabilitada”.

Isso quer dizer que nenhum procedimento deve ser tal que interrompa o funcionamento da tela e suas premissas como clique de botão, arraste da tela, minimização etc.

Para tanto o grupo deverá implementar uma solução utilizando seus conhecimentos de Thread em Java.

Os grupos devem efetuar uma implementação usando Thread ou Runnable e, em hipótese alguma, usar a classe Timer e nem TimerTask.

2 ENTREGA DO TRABALHO

O trabalho deve ser feito em grupos de, no máximo, 4 componentes.

- O grupo deve compactar o projeto gerado no netbeans ou os arquivos com código fonte (.JAVA)

A pasta do projeto JAVA (feito no NetBeans) deve ser compactada em um arquivo (ZIP).

Esse arquivo deve ser nomeado seguindo o seguinte critério:

- POO2_162_TR01_nome01_nome02_nome03.zip
 - (para grupos de 3 alunos)
- POO2_162_TR01_nome01_nome02_nome03_nome04.zip
 - (para grupos de 4 alunos)

Onde nome01 (nome02, nome03, nome04) são os nomes dos integrantes do grupo. Por exemplo, para os alunos Paulo Roberto, José Alcantara e Marina Ferreira o nome do arquivo será:

POO2_162_TR01_PauloRoberto_JoseAlcantara_MarinaFerreira.zip



O arquivo deverá ser postado no moodle (<http://moodlep.catolica-es.edu.br/moodle>) da disciplina por um integrante do grupo (APENAS). Grupos com mais de um envio não serão avaliados!

A data limite de envio é 05/09/2016 até às 23:55.

2.1 DICAS

- (I) Pesquise sobre o método `System.currentTimeMillis()` para proceder com a aquisição da diferença de horário do sistema.
- (II) Implemente uma classe adicional para tratar a escrita do cronômetro.

3 AVALIAÇÃO DO TRABALHO

Serão quesitos de averiguação para a nota do trabalho:

- **Pontualidade na entrega**
- **Conformidade com o solicitado no trabalho**

Este trabalho vale no máximo 50 pontos.

3.1 ATENÇÃO

O trabalho é em grupo e na data limite de envio os integrantes deverão estar aptos a responder os questionamentos do professor sob a pena de até 40% da nota. Serão feitas arguições a qualquer integrante do grupo e respostas erradas quanto ao apresentado no código acarretarão descortes na nota.

Dúvidas a respeito do trabalho serão sanadas nas aulas ou via e-mail.