

### Suporte para GitHub:

Carregando arquivos pelo navegador: [https://drive.google.com/open?id=1Klf0HCJcB\\_4Q5B7efMXrs\\_YYDXGwH89U](https://drive.google.com/open?id=1Klf0HCJcB_4Q5B7efMXrs_YYDXGwH89U)  
Tutorial Basico GitHub com Eclipse e EGit Usando Chave SSH: <http://www.youtube.com/watch?v=fFB5azTSGZw>  
Usando Github com Github Desktop em Projetos Eclipse: <http://www.youtube.com/watch?v=EgHljlYyS4U>  
Usando Github com SSH no Terminal Linux com chave gerada no Eclipse: <http://www.youtube.com/watch?v=0s699q5Sia4>  
Usando Github com SSH no Terminal Windows com chave gerada no Eclipse: <http://www.youtube.com/watch?v=DaydwPB2WSI>

### Vídeo Suporte:

Gerar Bibliotecas Java: [https://youtu.be/9x3\\_c\\_Oi6O0](https://youtu.be/9x3_c_Oi6O0)

Listas Dinâmicas: <https://youtu.be/H8D0LFN1uhs>

Gerando link compartilháveis: <https://drive.google.com/file/d/1cyoxa5W67MY5xDM6gCYpklel1GU3QCKa>

**Para todos os exercícios, quando solicitado teste de mesa, carregar a solução para um drive compartilhado e quando solicitado desenvolvimento, definir o que se pede e aplicar o código em Java e carregar a solução no Github.**

1. Simular o comportamento de listas dinâmicas para os algoritmos abaixo (A simulação deve deixar evidente a Lista que sobrou na memória):

a)

```
Para (i = 0 ; i < 10 ; i++) {  
    Se (i % 2 == 0) {  
        lista.addFirst(i * i);  
    } Senão Se (i <= 6) {  
        lista.addFirst(i);  
    } Senão {  
        escreva(lista.get(size() - 1));  
        lista.removeLast();  
    }  
}  
Escreva(Size());
```

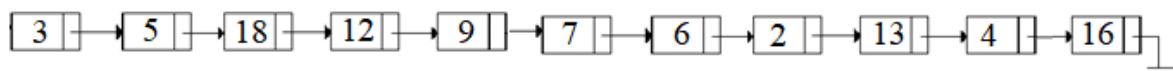
b)

```
Para (i = 100 ; i < 115 ; i++) {  
    Se (lista.isEmpty()) {  
        lista.addFirst(i + 100);  
    } Senão Se (Size() <= 4) {  
        lista.addLast(i + 50);  
    } Senão {  
        Escreva(lista.get(0));  
        lista.removeFirst();  
    }  
}  
Escreva(Size());
```

2. Em Java:

a) Ajustar o projeto de aula para criar uma biblioteca de Lista Genérica, gerando o JAR Lista

3. Dada a Lista L abaixo, fazer:



- Determine a sequência de passos para inverter os elementos da própria lista e fazer a exibição dos elementos
- Fazer, em Java, um novo projeto que insira a Lista L como acima e implemente operações que permita inverter os elementos da própria Lista e a exibição dos elementos

\* É permitida a utilização de TADs (Estruturas de Dados) auxiliares