

Nome:

Gustavo Leão Nogueira de Oliveira

Perguntas:

1. Qual a diferença entre as duas implementações apresentadas? Explique com suas o que cada uma delas realiza.

Ambas, apesar de parecer ser parecidas não são pois a que executa de forma paralela organiza como serão executadas as threads, e ja a segunda (de forma concorrente) embaralha a execução, sendo mais complexa de entender a saída.

2. Qual dos dois são concorrentes e/ou paralelo?

Concorrente: o [Projeto02](#), pois este executa todos os calculos, só que de maneira escalonada, ou seja, de maneira que cada execução por um tempo e variada.

Paralelo: já o [Projeto01](#), é paralelo pois é executado de forma que as Threads sejam executadas ao mesmo tempo.

3. Sabendo que um aplica o conceito de interface em Java, qual a melhor estratégia de implementação que você considera? Justifique sua resposta apontando as vantagens e desvantagens de cada uma.

Sabendo que a estratégia do [Projeto01](#) implementa interface, considera essa mais complexa, porém utilizaria essa quando necessito que as threads criadas executem de uma forma mais "organizada" e executem de forma paralela (juntas).

Vantagens: execução e saída mais organizada das threads.

Desvantagens: dificuldade na programação por ser necessário criar mais códigos para criar desta forma.

Ja a forma do [Projeto02](#) é interessante quando posso fazer as threads competirem entre si, não necessitando serem executadas juntas.

Vantagens: facilidade por ter um código mais simpes de ser feito.

Desvantagens: por serem executadas de forma aleatória, não é recomendado quando as saidas tem de ser organizadas, sendo mais complexa de entender a saída.

4. Quantas Threads são criadas em cada projeto? Justifique.

[Projeto02](#) no projeto 2 é criado apenas uma thread,e instanciada. também por que a classe **MeuThread** tem o metodo `run()`.

Já no [Projeto01](#) temos duas, uma na interface **MeuRunnable** e uma na classe **MeuThread**.
