

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**

**DO TRIÂNGULO MINEIRO – CAMPUS UBERLÂNDIA CENTRO**

**CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA INTERNET**

**ROBERTO EUGENIO DOS SANTOS**

**“Algoritmo Miller-Rabin”,**

**UBERLÂNDIA, MG**

**2019**

**ROBERTO EUGÊNIO DOS SANTOS**

**“Teste de Algoritmo Rabin”,**

Conceito do Miller-Rabin

Orientador: **Guilherme Moraes**

**UBERLÂNDIA, MG**

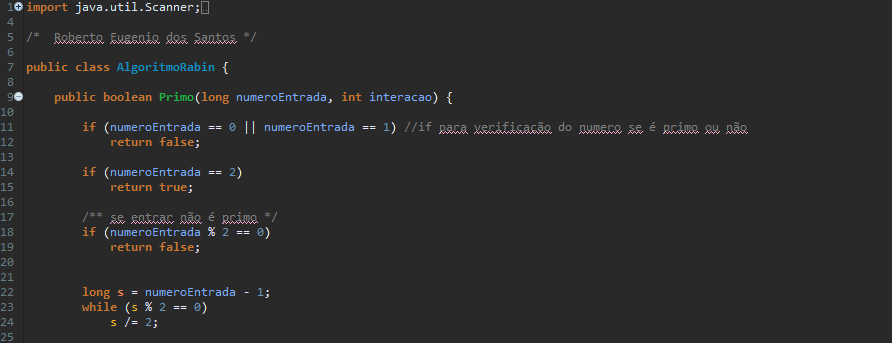
**201****9**

# Conceito Miller-rabin

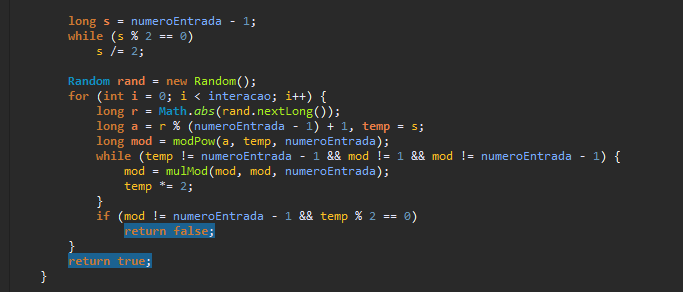
Nesse caso é um teste de probabilidade número *n*. Se um número *n* não passar pelo teste, *n* com certeza é um número composto, isso quer dizer que ele é um número que não é primo, e em contra partida se ele passar no teste a probabilidade dele ser primo, lembrando que é uma probabilidade.

A importância desse algoritmo se deve à assimétrica, porque precisa de uma garantia que uma grande quantidade de números primos grandes é vital para a segurança dos algoritmos. Esses números são tão grandes que testes não probabilísticos como o da simples divisão por números primos menores que o número gerado ou o tabelamento de todos os números primos são impraticáveis, este é o principal motivo de ser usar o teste de Miller-Rabin, ou **Rabin-Miller,** como às vezes também é chamado, esse teste é o mais utilizado para descobrir se é primo ou não.

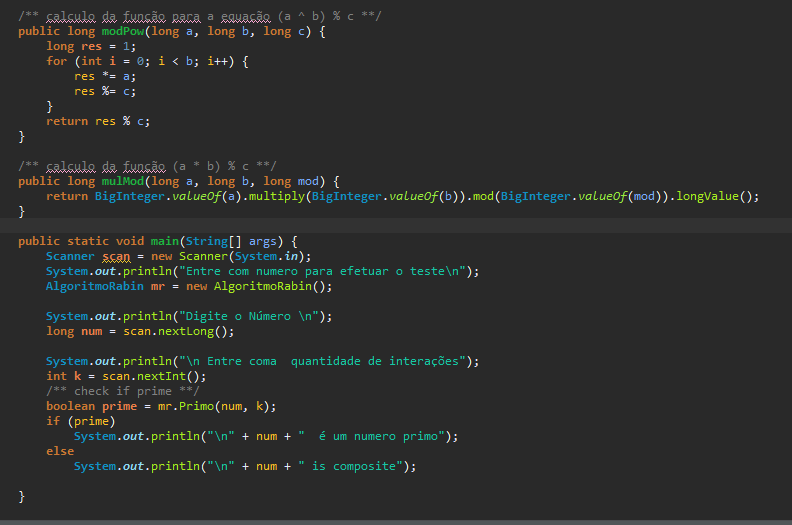
# exemplo código em java



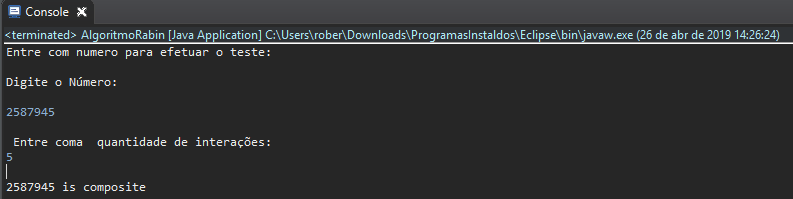
Verificação para saber se é primo ou não.



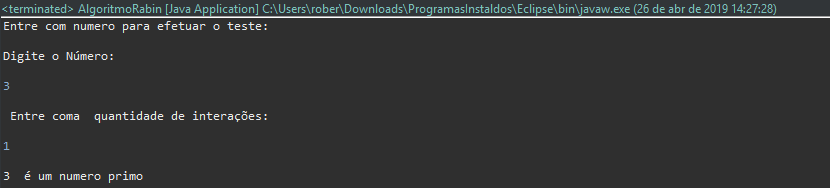
Verificação da quantidade de interação quanto mais interação maior a probabilidade ser número primo.



Calculando o número pra saber se ele tem a probabilidade de ser primo, ou se é um número não primo.



Neste exemplo usei um número maior para saber se ele é primo ou não e assim usando a quantidade de interações de cinco vezes, sendo ele um número não primo.



Este exemplo usou o número 3 para ser identificado como primo