

The background features a dark gray field with several curved lines. At the top, there are five white arcs of varying lengths and radii. Below these, a thick red arc spans across the upper middle. In the bottom left, there is a thick red arc and a white arc. In the bottom center, there is a long white arc and a red arc. In the bottom right, there is a thick red arc and a white arc.

Insper **Supercomputação**

Projeto 1 - Otimizações em CPU

- Algoritmo de mineração de criptomoedas implementado em C++
- Código inicial propositalmente ineficiente (gargalos e mau uso de memória).
- Identificar problemas, propor hipóteses de melhoria e aplicar otimizações.
- Mensurar o impacto das mudanças no desempenho (tempo, throughput, uso de CPU/memória).
- Produzir relatório técnico documentando todo o processo de análise e otimização.

Projeto 1 - Otimizações em CPU

- A dificuldade da mineração varia pelo número de zeros no início do hash.
- Aumentar a dificuldade eleva o custo computacional, exigindo melhores decisões de otimização.
- Planejar e executar testes considerando os recursos e filas do Cluster Franky.

Bloco #1

✎ Lícia (Insper)

✓ 28/08/2025, 11:30:09

nonce: 16012

hash: 00000cea0aa5ba8e9758a39773aaa75466
9a99e047411e3cb0594d99434e9234

dificuldade: 5

O que é um bloco?

- Cada transação é registrada em um “bloco”

Blockchain

Bloco #1

🔗 Lícia (Insper)

✅ 29/08/2025, 08:09:46

dificuldade: 5

[Clique para ver/fechar detalhes](#)

Bloco #2

🔗 Lícia (Insper)

✅ 29/08/2025, 08:10:51

dificuldade: 6

[Clique para ver/fechar detalhes](#)

Bloco #3

🔗 Lícia (Insper)

✅ 29/08/2025, 08:12:30

dificuldade: 7

[Clique para ver/fechar detalhes](#)

Bloco #4

🔗 Lícia (Insper)

✅ 29/08/2025, 08:17:06

dificuldade: 8

[Clique para ver/fechar detalhes](#)

Bloco #5

🕒 aguardando minerador

🕒 29/08/2025, 08:17:06

dificuldade: 7

[Clique para ver detalhes](#)

Cada bloco tem uma “chave” única (hash)

- Cada bloco no Bitcoin tem um cabeçalho com várias informações
- Esse cabeçalho é passado por uma função de hash criptográfica (SHA-256 duas vezes).
- Para o bloco ser aceito pela rede, o hash resultante precisa ser menor que um valor-alvo definido pela dificuldade da rede (ou seja, o hash deve começar com um certo número de zeros).
- O nonce é o campo que os mineradores mudam repetidamente dentro do cabeçalho para gerar novos hashes até encontrar um que satisfaça a condição.

Bloco #1

↗ Lícia (Insper)

✓ 28/08/2025, 11:30:09

nonce: 16012

hash: 00000cea0aa5ba8e9758a39773aaa75466
9a99e047411e3cb0594d99434e9234

dificuldade: 5



O minerador procura nonces

- O desafio do minerador é achar um número (o nonce) que revela o hash do bloco, que sempre vai começar com um certo número de zeros.
- Os zeros indicam a nível da dificuldade para validar aquele bloco
- O sistema se adapta a velocidade de validação dos blocos pelos mineradores na rede.
- Com esse sistema fica difícil falsificar informações nos blocos.

Bloco #1

🔗 Lícia (Insper)

✅ 28/08/2025, 11:30:09

nonce: 16012

hash: 00000cea0aa5ba8e9758a39773aaa75466
9a99e047411e3cb0594d99434e9234

dificuldade: 5

A Lógica do Blockchain

Cada bloco carrega informações sobre o bloco anterior

Bloco #1

🔗 Lícia (Insper)

✅ 29/08/2025, 08:09:46

dificuldade: 5

nonce: 48247

hash: 00000394164af3c0438a4e298505af420e
f7a31a12d46a6ce819306e2c2bfad5

💰 tx: Lícia → Prog de Bolsas Insper (R\$ 33,00) •
prev 000000000000...00000000 • hash
00000394164a...2c2bfad5

[Clique para ver/fechar detalhes](#)

Bloco #2

🔗 Lícia (Insper)

✅ 29/08/2025, 08:10:51

dificuldade: 6

nonce: 15035

hash: 0000009215d10d79663b08d56885e6e9ac
e76c0677cfb5b77b40ea6fb5f29eb1

💰 tx: Lícia → Prog de Bolsas Insper (R\$ 36,00) •
prev 00000394164a...2c2bfad5 • hash
0000009215d1...b5f29eb1

[Clique para ver/fechar detalhes](#)

Criando uma grande cadeia de registros

Rúbrica de Avaliação

Conceito	Critérios Técnicos
C	Executa o minerador sequencial no cluster Franky com dificuldade 6 zeros , Realiza Passagem de objetos grandes por referência ou ponteiro; Minimização de cópias desnecessárias; Uso eficiente de buffers; Implementa uma heurística eficiente.
B	Executa o minerador com dificuldade 7 zeros , Realiza as otimizações da rúbrica C e aplica paralelização com OpenMP ou distribuição com MPI
A	Executa o minerador com dificuldade 8 zeros , Realiza as otimizações da rúbrica C e aplica paralelização com OpenMP E distribuição com MPI

Projeto 1 - Otimizações em CPU

Grupos de até 3 alunos - Entrega 29/Setembro

- Espera-se que o aluno entenda os problemas do código base antes de corrigir.
- O binário `transações` simula um ambiente de rede assíncrono com envio de blocos não simultâneos, para que o código minerador seja tolerante a ordem e tempo de chegada.
- A execução dos testes pode demorar minutos. Se planeje bem, escolha adequadamente a fila que será utilizada nos seus testes.

Bora ver como isso funciona!



<https://18-118-206-124.sslip.io/>