RELATÓRIO DE RESULTADOS - PROJETO LLM TOOLS

Aluno: Gabriel Batistuta

Critérios de Aceitação:

- Prompt 1 — Transferência com possível taxa:

Prompt	Cenário	Operações
1	A	withdraw(BC12345, 1000.0) deposit(ND87632, 1000.0) taxes(BC12345, 1.50)
1	B	withdraw(BC12345, 1000.0) deposit(ND87632, 1000.0) -> FAILED returnValue(BC12345, 1000.0)

- Prompt 2 — Retiradas repetidas e depósito final com taxa:

Prompt	Cenário	Operações
2	A	withdraw(BC3456A, 500.0) withdraw(BC3456A, 500.0) withdraw(BC3456A, 500.0) withdraw(BC3456A, 500.0) withdraw(BC3456A, 500.0) deposit(FG62495S, 2500.0) taxes(FG62495S, 250.0)

2	B	withdraw(BC3456A, 500.0) withdraw(BC3456A, 500.0) withdraw(BC3456A, 500.0) deposit(FG62495S, 1500.0) taxes(FG62495S, 150.0)

- Prompt 3 — Dupla retirada condicional e depósito/pagamento

Prompt	Cenário	Operações
3	A	withdraw(AG7340H, 600.0) withdraw(TG23986Q, 700.0) deposit(WS2754T, 1300.0) payment(WS2754T, 1200.0)
3	B	withdraw(AG7340H, 600.0)->FAILED withdraw(TG23986Q, 700.0) returnValue(TG23986Q, 700.0) OU withdraw(AG7340H, 600.0) withdraw(TG23986Q, 700.0)->FAILED returnValue(AG7340H, 600.0)

AVALIAÇÃO DO MODELO MISTRAL:LATEST

Obs: como o modelo às vezes parava do nada implementei uma forma de pular os cenários já rodados anteriormente, talvez esteja mal implementado pois não tenho muita familiaridade com a linguagem java

Rodar do início (executa todas as configs/prompts/cenários):

```
mvn exec:java
-Dexec.mainClass="br.university.project.runner.MainRunner"
```

ou explicitamente com args vazios:

```
mvn exec:java
-Dexec.mainClass="br.university.project.runner.MainRunner"
-Dexec.args=""
```

Rodar a partir de um caso específico (ex.: CONF3 P3 A) — executa apenas a partir daquele cenário em diante:

```
mvn exec:java
-Dexec.mainClass="br.university.project.runner.MainRunner"
-Dexec.args="CONF3 P3 A"
```

TABELA DE RESULTADOS - MÉTRICAS AGREGADAS

Config	Prompt	Cenário	Corretude	Consistência	Abordagem	Erros
CONF1	P1	A	(100.0%)	Não	BankToolsA	0
CONF1	P1	B	(0.0%)	Sim	BankToolsA	0
CONF1	P2	A	(0.0%)	Sim	BankToolsA	0
CONF1	P2	B	(0.0%)	Sim	BankTool	0

					sA	
CONF1	P3	A	(0.0%)	Sim	BankTool sA	0
CONF1	P3	B	(0.0%)	Sim	BankTool sA	0
CONF 2	P1	A	(0.0%)	Sim	BankTool sB	0
CONF 2	P1	B	(0.0%)	Sim	BankTool sB	0
CONF 2	P2	A	(0.0%)	Sim	BankTool sB	0
CONF 2	P2	B	(0.0%)	Sim	BankTool sB	0
CONF 2	P3	A	(0.0%)	Sim	BankTool sB	0
CONF 2	P3	B	(0.0%)	Sim	BankTool sB	0
CONF3	P1	A	(100.0%)	Sim	BankTool sB	0
CONF3	P1	B	(0.0%)	Sim	BankTool sB	0
CONF3	P2	A	(0.0%)	Sim	BankTool sB	0
CONF3	P2	B	(0.0%)	Sim	BankTool sB	0

ANÁLISE DETALHADA DE CASOS PROBLEMÁTICOS

CASO PROBLEMÁTICO: CONF1-P1-A

- Corretude: 10/10 (100.0%)
- Consistência: Não
- Abordagem (Ferramentas): BankToolsA

- Execuções com erro: 0/10
- Uso de ferramentas: BankToolsA: 10

CASO PROBLEMÁTICO: CONF1-P1-B

- Corretude: 0/10 (0.0%)
- Consistência: Sim
- Abordagem (Ferramentas): BankToolsA
- Execuções com erro: 0/10
- Uso de ferramentas: BankToolsA: 10
- Execuções incorretas (lógica errada): 10
  Exemplos de falhas:
  Execução b486548a: operações observadas: withdraw(BC12345,1000.0), deposit(ND87632,1000.0)->FAILED, taxes(BC12345,1.5), withdraw(BC12345,1.5)
  Execução 62f80d92: operações observadas: withdraw(BC12345,1000.0), deposit(ND87632,1000.0)->FAILED, taxes(BC12345,1.5), withdraw(BC12345,1.5)

CASO PROBLEMÁTICO: CONF1-P2-A

- Corretude: 0/10 (0.0%)
- Consistência: Sim
- Abordagem: BankToolsA
- Execuções incorretas: 10
  Exemplos: apenas withdraw(BC3456A,500.0) executado, diversas operações esperadas ausentes.

CASO PROBLEMÁTICO: CONF1-P2-B
Sem variações relevantes em relação ao caso anterior.

CASO PROBLEMÁTICO: CONF1-P3-A

- Corretude: 0/10

- Operações observadas: withdraw(AG7340H,600.0), withdraw(TG23986Q,700.0)

- Operações esperadas ausentes: deposit(WS2754T,1300.0), payment(WS2754T,1200.0)

CASO PROBLEMÁTICO: CONF1-P3-B

- Mesmo padrão do caso A, porém com uma operação falhando e outra executando.

Os casos problemáticos para CONF2 seguem o mesmo padrão:

- Corretude sempre 0/10

- Consistência sempre Sim

- Abordagem sempre BankToolsB

- Diversas operações esperadas ausentes, operações executadas inconsistentes

CASO PROBLEMÁTICO: CONF3-P1-B

- Corretude: 0/10

- Uso simultâneo de BankToolsA e BankToolsB

- Falha lógica na etapa de retorno de valor após erro de depósito

CASO PROBLEMÁTICO: CONF3-P2-A e CONF3-P2-B

- Mesma lógica incorreta observada nas versões correspondentes de CONF2

ANÁLISE ESPECÍFICA - FALHA DO MODELO MISTRAL NO CONF3 P3 A

PROBLEMA IDENTIFICADO:
O modelo Mistral começou a falhar na configuração CONF3, Prompt P3, Cenário A, gerando parâmetros inválidos para as ferramentas.

DETALHES DO ERRO:

- O modelo enviou JSON malformado

- Tentou usar condicionais dentro dos parâmetros, como: {"if": "$[0].success && $[1].success"}
- Isso é incompatível com LangChain4J

CONTEXTO DA FALHA:

- Antes de CONF3 P3 A: execuções bem-sucedidas
- A partir deste ponto: falhas consistentes
- Prompt P3 contém lógica condicional mais complexa
- CONF3 oferece múltiplas ferramentas, aumentando a complexidade da decisão

IMPLICAÇÕES PARA O PROJETO:

- A falha confirma limitações do modelo Mistral para prompts complexos
- Reforça a necessidade de avaliar diferentes LLMs com as mesmas APIs
- O erro é um resultado útil para comparação entre modelos