Title: getPoolReserves() returns token reserves in incorrect order

Bug Description
The getPoolReserves() method returns token reserves in an order that does not match the token pair defined in the pool metadata. This leads to incorrect price and slippage calculations in applications relying on the SDK.

Steps to reproduce
Initialize the SDK and connect to the mainnet.
Retrieve a known pool e.g. "SOL-USDC".
Call getPoolReserves() on the pool.
Compare the returned reserves with the expected token order from the pool metadata.

```
const sdk = new SarosSDK(connection);
const pool = await sdk.getPool("SOL-USDC");
const reserves = await pool.getPoolReserves();
console.log(reserves);
```

Expected Behavior
Reserves should be returned in the same order as the token pair defined in the pool metadata:
[SOL_reserve, USDC_reserve]

Environment

Component	Version
SDK Version	1.3.2
Node.js	18.17.0
OS	MacOS 13.5
Network	Mainnet

Severity
High - This issue affects core financial calculations and may result in incorrect pricing or slippage estimates in production environments.

Additional notes
The pool metadata correctly identifies tokenA as SOL and tokenB as USDC.
The issue appears to be consistent across multiple pools.
A fix should ensure that getPoolReserves() respects the token order defined in the pool metadata.