Programming Task

Code Explanation

1. **Hexadecimal to Byte Vector Conversion:**
   - The main function starts by defining a hexadecimal string representing a serialized Bitcoin transaction.
   - This hex string is converted to a vector of bytes (std::vector<uint8_t>). This is done in a loop where each pair of hex characters is converted to a byte.

2. **Deserialization Function:**
   - The **deserializeTransaction** function is called with the byte vector.
   - It reads and parses the transaction data step by step.

3. **Version:**
   - The first 4 bytes of the transaction represent the version number.
   - This is extracted and printed.

4. **Input Count:**
   - The number of transaction inputs is read using the readVarInt function, which handles the variable-length integer format used in Bitcoin transactions.
   - This count is printed.

5. **Transaction Inputs:**
   - For each input, the following fields are extracted:
     - Previous transaction hash (prevTxHash)
     - Previous transaction output index (prevTxIndex)
     - Script length and script signature (scriptSig)
     - Sequence number (sequence)
   - These fields are printed in a formatted manner.

6. **Output Count:**
   - The number of transaction outputs is read and printed similarly to the input count.

7. **Transaction Outputs:**
   - For each output, the following fields are extracted:
     - Value (amount in satoshis, converted to BTC for display)
     - Script length and script public key (scriptPubKey)
   - These fields are printed in a formatted manner.

8. **Lock Time:**
   - The final 4 bytes represent the lock time of the transaction.
   - This is extracted and printed.

Input

The input is a hexadecimal string representing a serialized Bitcoin transaction:

```
"020000000001056a51edb2fdd26ff90ef8a086fd1ec0c63210d0edfa67397f0dbeeeb621364e330100000000feffffff2577513282bb52facf186cf6f8893d69aae1
cdb2b0eb0317fc8794f4819af2b5010000006a47304402200607b736df8fe395d861a0754c77eb71ead51eb4a5078d93f83861e163519798022059a8c9d34af3b9632
b119c68dc007dcdac3001eb358925a1db68eeac370d3d4b01210233541665c6963ea1b867f8b0076b241e87f7d48f3afa0566f4f22b753d36da32feffffffd6b5eaf
d8070281dd36c4981000aac89bab0a54af7a1ca052ff9e895f7443d30100000000feffffff0c8e8cbad2a4feab6875454f1ece3fd33afb333d1624350f2636098cc3c
cf3680100000000feffffffc64d43e3bb402aeba7cd58ba084140d87f8c8062ec1c5e668d0695533cd776d4000000006a473044022070dea539e1f01589cb87926ef8
646540e453a5c90b80907680fefcbd513767d4022035ebc67a4360d62b9d1bdb2d89f210c78d7bc4316eacad1a4fd89427fe001ece0121024e61652d99350a41b7395
a12263bdb1e350aca7f6e884733d8ef5260e833ef49feffffff0226d6fd000000000017a914978e3e09c10b72077c2102d1853883a5a4f0b61b87b74b150000000000
160014f7533a613e95c63be15a519b8327e7f33ac126a60247304402206ecad418dbf2ab3dc6903ee44e3077e058df9968241fdf931093fbaf9a51e11b02201494d62
531e6773220072856fc9726bc1e5c93767ea98da4b75d51411187cb280121039350003ee10d39a0811696f7deccf4c31c924ce55c2506191e819833f6c6baa3000247
3044022021962ffa5db42745d5a9a851fc508fab603ca690d17ae962a3aa9a6f0ff9d9ed02204b377666ab296cef8ff5abe9bb35eabd714f8845dbac9f352bcf0af89
a8c220a012102a3338fc7a90f8e3802d61f1ea73d77edd257ba97848e3d3a97d790eb3105845c0247304402200de7b1f320d0498f2053e7c46bad62bf5b58b87f5b1f
96d64ffa007afa8c0ba6022056669ccfaf4772934cd264755edb94228387802ca038e238977ca5a4dea6d4260121026491015716a9c02db53743dcfa33bd01a45fb28
7e092f09e5d33756f4c9b77040086c70c00"
```

Output

The output represents the parsed information from the Bitcoin transaction:

```
Version: 2
Input Count: 1
Input:
  TXID: 64e32162ebeeebdf9736fadf0e0d2163c6c0ed86108a0eff96fd2fdb21ed516a
  Vout: 330014365
  ScriptSig: 47304402200607b736df8fe395d861a0754c77eb71ead51eb4a5078d9383861e163519798022059a8c9d34af3b9632b119c68dc007dcdac3001eb3589
  Sequence: 4294967294
Output Count: 2
Output:
  Value: 0.02410081 BTC
  ScriptPubKey: 76a914978309c10b72077c2102d1853883a5a4f0b61b87b74
Output:
  Value: 0.06827159 BTC
  ScriptPubKey: 76a914f7533a613e95c63be15a5198327e7f33ac126a6088ac
Lock Time: 0
```

Explanation of Output

- **Version:** The transaction version is 2.
- **Input Count:** There is 1 input in the transaction.
- **Input Details:**
  - **TXID:** The transaction ID of the previous transaction output being spent.
  - **Vout:** The index of the specific output in the previous transaction.
  - **ScriptSig:** The script signature used to prove ownership of the input.
  - **Sequence:** The sequence number of the input.
- **Output Count:** There are 2 outputs in the transaction.
- **Output Details:**
  - **Value:** The amount of Bitcoin being sent (in BTC).
  - **ScriptPubKey:** The script public key, which specifies the conditions under which the output can be spent.
- **Lock Time:** The lock time is 0, indicating the transaction can be included in a block immediately.