1)What are the concepts (borrowing, ownership, vectors etc)

2)What is the organization?

3)What is the contract doing? What is the mechanism?

4)How could it be better? More efficient? Safer?

Acct Check Native

1)

Library imports, macros, mod function, public functions and structs are used.

To use the anchor lang (Solana framework), they used use anchor\_lang::prelude and the declared the ID. After importing anchor library, the public ID was declared with declare\_id!() function.

On macros, #[program], #[derive(Accounts)], #[account(mut)] were used and the organization of macros start with #[program] with a public function, after the public function, #[derive(Accounts)] has been used and a public struct was used with parameters #[account(mut)] macros.

2)

The program is organized by macros, mod function, public functions and structs.

3)

The contract allows native accounts to be created with limited lines of code, payers sign the information, check accounts if owners have accounts, change accounts when seen fit by the owners.

4)

The code would be more efficient if more public functions can be created to accommodate changing accounts. In addition to the initial recommendation, due to the sensitive function of the program, security measures can be implemented.

CPI NAT HAND

1)

The concepts of the code are macros, public functions, structs, mod function, Boolean variable, mutable variables and match function.

Structs: Multiple public structs have been used independently and all actually started as soon as the accounts have been derived. Similar to classes in python, each struct individually represent the behavior of lever initialization, setting up power status and the current status with a Boolean variable to check whether power is on or off.

Functions: Public functions have been used to initialize and switching power, the latter has a match function that is a conditional to check power status. Mod is used in an appropriate manner as there were no function or struct could be initiated and Boolean variables checked whether statements were true. Finally, mutable variables were simply parameters.

Macros: Macros are derived before each function and struct in Rust, which is common with anchor.

2) The code is organized to library imports, id declaration, mod declaration, functions and structs.

3)

The program serves as a hand for the extended lever contract. It pulls the lever, switches the power, and pulls the lever with all these simple commands.

4)

It can allow the lever to be pulled to close, which was not existent.

CPI NAT LEVER

1)

The concepts of the code are mod, functions, match and conditional functions, structs and macros.

2)

Each line of function/structs begin with macros to initialize accounts and the mod is used with public functions, hosting two functions at the same time with mutable statements and conditionals. At the final part of the code, structs are used to initialize the lever.

3)

The code pulls the lever and checks power status.

4)

It could allow power to be shut down.