

## Refactor

The refactoring process was rather simple. My goal was simply to extract the code block for each card and place it in its own separate function without changing anything, so it made sense to pass everything by pointer. The cards I refactored were **Adventurer**, **Smithy**, **Village**, **Council Room**, and **Great Hall**. To perform the refactoring process, I simply took note of the variables used in the card's switch case and passed them in via the function call. As stated before, everything was passed by pointer in order to preserve any existing or future dependencies in the cardEffect function. As every switch case returned a 0, I made every function do the same. In cardEffect, I made the function calls also serve as return statements, e.g. return smithyCard(<parameters>). I refactored each card individually and checked to see whether the program compiled and ran properly following each process.

## Bugs

**Adventurer:** Line 647: changed `while(*drawntreasure<2)` to `while(*drawntreasure==2)`

**Smithy:** Line 697: commented out the discardCard function call

**Village:** Line 709: changed the trash flag from 0 to 1, making the discardCard function think that the card is trashed, not just discarded

**Council Room:** Line 679: instead of checking to see if iterator i refers to players not currently taking their turns, it now checks to see if i refers to the player whose turn it is (changed `if(*i != currentPlayer)` to `if(*i == currentPlayer)`).