



✉ Pakistan@blockstack.com

☎ +92 334 4775648

📍 Headquartered in New York City

# Assignment 1

Topic: Merkel Audit

## Problem Statement

You have to initialize a txt file containing only binary digits (0 and 1). Make sure the no of bits is in the power of 2. You have to implement Merkel Audit in this assignment. Chunks should be in Even numbers. You can also initialize the character array and append 0 or 1 to mimic the bits from a file. The next step will be is to divide into chunks.

```
class Chunk{
    char *chunk; // This array will contain bits like 000100101010101
    char *hash; // Hash of the chunk , you may implement any good hash function.

    char* CalculateHash(); // Calculates the hash of the chunk
}

class File{
    int ChunkSize;
    Chunk *AllBits;
    char* MerkelRoot;

    void ChunkifyFile(); // Populates AllBits array with Chunk Object.
    char* CreateMerkelTree(Chunk*AllBits); // Creates MerkelRoot.
    void ModifyRandomChunk(); // This function will modify Single Chunk in AllBits array.
    int FindDirtyChunk(); // This will create the Merkel Tree again and will find the
                          // dirty/Modified Chunk in Top Down approach as studied in class.
}
```

You will have to add functionality for storing and creating Merkel tree.

1. For this first you will read the whole file in Chunks in predefined Chunk array.
2. Chunkify the array.
3. Create Merkel Root only.
4. Now Modify any Chunk by flipping some bits of it.
5. Again, recompute Merkel Root and store it into another variable
6. Compare it with old Root, if it is not same then move in top-bottom approach to find the dirty chunk. For this you will have to create Merkel Tree not just Merkel root.
7. Report the Modified Chunk and print both new and previous Chunks.

Note (Structure provided is just for the understanding)

O  
\\/  
| Happy Coding ☺  
/\



www.pakistan.stacks.co Find us on

