

SF Lab - 2
Assignment - 1

Anuj M. Choure
19CS02010

Pseudo Code (Encoding):

```
audioFile = read(filePath)
Key = input()
message = input()
message += "$"
message = convertToBinary(message)
msgLen = len(message) + 16
message = convertToBinary(msgLen) + message
range = len(audioFile) // len(message)
a = getStartIndex(key)
step = getStep(key)

strIdx = 0
i = a

while strIdx < 16:
    changeLSB(audioFile[i], message[strIdx])
    strIdx += 1
    i += 1

while strIdx < len(message):
    changeLSB(audioFile[i], message[strIdx])
    strIdx += 1
    i += step
    getNextRandomStep(step, range)
```

Pseudo Code (Decoding):

audioFile = read(filePath)

Key = input()

a = getStartIndex(key)

step = getStep(key)

i = a

msgLen = ""

while len(msgLen) < 16:

 msgLen += audioFile[i]

 i += 1

range = len(audioFile) // msgLen

getNextRandomStep(step, range)

extractedMsg = ""

while message:

 extractedMsg += audioFile[i]

 i += getNextRandomStep(step, range)

Algorithm (encoding):

1. Reading the file
2. Getting start index and initial step from the user given key
3. Finding the distance between bits storing locations, such that the secret message is distributed throughout the audio.
4. Inside a while loop change the LSB of the frame
5. While the message is not stored, set the step as a random seed and generate the next random step and repeat step 4.

Algorithm (decoding):

1. Read the file
2. Get the start index and initial step from the user given key.
3. Get the message length from the initial 16 bits, which are stored continuously.
4. From the message length, get the range such that the message is distributed throughout the audio.
5. Using the range and step generate the sequence of random numbers and extract the message bits till the delimiter is reached.