MODBENDER INSTRUCTIONS

1. program start

run the program on terminal:

```
python3 modbender.py input.wav output.wav [options]
```

2. load input file / set output file name

input file “input.wav” is the 16-bit PCM WAV file that will be processed
output file “output.wav” is the WAV file that will be created
input file must be located in the script directory

3. parameters

--lsb
enables/disables LSB manipulation
“--lsb n” disables it
“--lsb y random” enables random LSB processing
“--lsb y markov” enables markov-based LSB processing

--xor
enables the XOR operation. allowed values are “y” or “n”

--patterns
hexadecimal XOR patterns separated by spaces as: “0xAA 0xFF 0x13 0x7C”

--markov
enables or disables markov-based window redistribution. allowed values: “y” or “n”

--bmarkov
enables or disables byte-level markov processing. allowed values: “y” or “n”

--bwindow
window size for byte-level markov processing. examples include 4, 8 or 16

--mwindow
fixed window size for audio markov reordering. examples include 512 or 1024

--xorwindow
fixed window size for modulated XOR. examples include 512 or 1024

--markovorder
order of the markov model for audio reordering or byte-level markov processing. examples: “2” or “3”

--bmarkovorder
order of the Markov model specifically for byte-level markov processing. examples: “2”, “3”

--smooth
smoothing factor for mwindow and xorwindow. a value of 0 means no smoothing. 1 means maximum smoothing

--rate

sample rate of the output WAV file. example: “44100”

--iterations (or -n)
number of times the entire processing pipeline runs. 1 by default

--order
defines the order in which the operations are pipelined. available values are:
l - LSB
x - XOR
m - sample-level Markov redistribution
b - byte-level markov redistribution
default order being “b l x m”
example of a custom order: “l x m b”

4. complete example

```
python3 modbender.py input.wav output.wav --lsb y markov --xor y --patterns 0xAA
0xFF 0x0F 0x55 --markov y --bmarkov y --bwindow 4 --mwindow 1024 --
xorwindow 1024 --markovorder 3 --bmarkovorder 2 --smooth 0.1 --rate 44100 --
iterations 2 --order b l x m
```