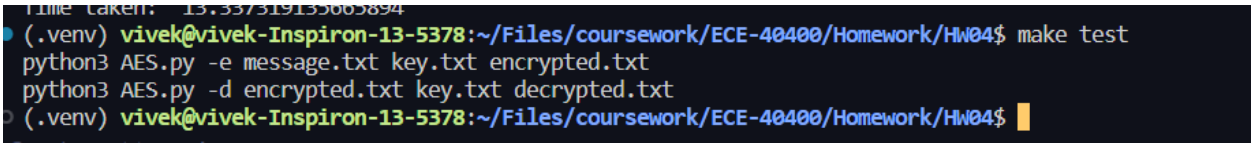
Homework 4 – AES

rpanchag@purdue.edu
ECE404

2/13/2024

Homework 4 Explanation and output:

Preamble: This implementation makes heavy use of lecture code given in the .py files that are downloadable from Professor Kak's website. The functions taken from lecture code are listed in my breakdown of each function.



Commands that produced the required output.

Encrypted output:
3ba1ab4b7fe412ca26c7a25cff913d1b748da805c97c83554d9e9cf5b12243ff03a8c6b6dcbc520750a14df9
b646fa480d1e64cc2e9174a23dbed6aad77144350ff768093cf7571852a26ffa36fe47652a546acf9d4bc1ad
395a92553b4b7e0a5a7811d7b95d95cacc117e344ac093da247168cd4bbbda5bc2866fd044c8ca18ecd2b
6a78bfe19520f22b7fa12862132e32ee78c5e4200166c40f1a93f9b08c5f67b9bde38d34ed34bd03183a52
9a5a62d81b1cf084832fcb9139a51100a04c7c631d3fbfa5bb9b8cbe970f02213ab07d3e179313142865fb8
b022241552567964250cfa2aa97c59223d30a2a7da8974d0f6c34f4f46ed6cab53e483f95d4ed157bb78ce
078a88397c9d656830fadd080d729ac7428a6ca3c17ad67d0cf16d35a8ecb35cd818a380309332c4cc29d0
0b6fe542b67724295b49804b2122b5b24e6f09e22451bb77c6876d51b7294b405dcff0cdc83754538442fc
c766bfe4fac839e932f757aebbe7f43c87d08249c6ef50d9adefa8eca175785ba0dbc31e2e61ba32a75f596
894ea736bcea8f351d3c4574539e7ad760c4a0c4b252e2dbc859c4b0a6b44fbf29b3fa7fddeace3855c6751
30ef65d4fa7f8125d4575f329cc93d75d14fdcb1419678cae4d686d4b72f56ac4d7974e3b1f1bbb3776dda5
db94b7d2ef1f73f96f7b24378a1e299271006cd478bd84fe7a24c67794e663668c918bdb65097099351e1e
bf6e7d1148754f1051d33156e4fb7e96cce8f976f6a0ad71d12b10d1b43458c02002bf1fc14c9c63e9033df
dcbc9baae76efc8e12a850fdd21ead4e9b14fb359a27fc4943b0d76714

Decrypted output:

Newly re-signed McLaren driver Lando Norris is confident that the team will be in the mix for race victories in 2024, but the Briton feels he may have to wait a little longer for a championship challenge. McLaren caught the eye last season by going from struggling to score points to regularly fighting for podiums, with highly effective upgrades being implemented following a technical reshuffle. Norris came close to scoring McLaren's first Grand Prix win since 2021 on several occasions, taking six P2 finishes, while team mate Oscar Piastri managed to triumph in the Qatar Sprint Race.

Code Explanation:

- Class AES()
  - __init__ (inside the AES class)
    - Generates round keys using gen_key_schedule_256 and gen_round_keys
  - encrypt (inside the AES class)
    - Opens the message file, and reads the file 128 bits at a time ensuring padding where needed.
    - It then runs the “pre round” xor with the first roundkey before doing all but the last round of encryption in a loop. It performs it in the order of, subbytes, shiftrows, mixcols, and add roundkeys. It then does the final round without a mixcols step and writes the bv to a file in hex.
  - decrypt (inside the AES class)
    - Opens the encrypted text file and ingests the entire hex of the file into a bv. This uses a counter variable to keep track of where in the large BV we are. I couldn’t figure out why using the bv.more_to_read method that I used in encrypt wouldn’t work here so I stuck to this method, which feels a lot more clumsy to me, but if it works it works :/. The commented code is reflective of my previous attempt.
    - I also flip the order of round keys to be used in the actual decryption step.
    - The script then performs the decryption steps, which are the xor with the roundkey[0] first, then the next few rounds of inv_shift_rows, inv_sub_bytes, add_round_key, and inv_mix_col. The last step is done separately to ignore the use of inv_mix_col, and then it is written in binary to the supplied file.
- gen_round_keys
  - gen_round_keys function taken from lecture notes. The function is fixed to use 14 rounds as per the directions for a 256 bit key.
- genTables
  - genTables function to generate substitution tables taken from lecture notes
- gen_key_schedule_256
  - key schedule generator function taken from lecture notes
- gee
  - G function taken from lecture notes
- gen_state_array
  - generates a 4x4 state array and populates it with the relevant parts of the input, which is the current working 128 bit cipher block.
- sub_bytes
  - Substitutes each item in the state array with the respective value from the subbytes table.
- inv_sub_bytes
  - Substitutes each item in the state array with the respective value from the inv_sub_bytes table.
- shift_rows

    - The first row isn't shifted, row 1 is shifted 1 byte to the left, row 2 is shifted 2 bytes, where row 3 is shifted 3 bytes.
- inv_shift_rows
    - Same thing as shift rows but shifts to the right
- mix_columns
    - This function performs a multiplication with a fixed polynomial as listed in the code in GF(2^8) using gf_multiply_modular in the Bitvector class to "mix up" the columns and introduce further confusion into the encryption.
    - The function itself first convers each element into bitvectors, then creates a deep copy into a new matrix, and uses the new matrix to hold the initial values of the old matrix while doing the multiplication.
    - The function then turns each element back into intvals and finally sends the result back.
- inv_mix_columns
    - Same thing as mix_cols except it uses the multiplicative inverse of each of the elements used in mix_cols to undo the mix_cols step.
- get_bv_from_state_array
    - iterates the state array and converts it back into a bv.
- add_round_key
    - xor step made easy as a function, takes a state array, converts it to a bv, and xors it with a roundkey before returning it.

Main:

Main just checks for the correct number of arguments, initializes the AES object, and calls encrypt or decrypt based on the input flag.