HW07 ECE 40400

Input

Boku no Kokoro no Yabai Yatsu is the greatest romance, slice of life manga I've ever read. It is a series of constant progress that respects the reader's time and trusts them to read between the lines - Characters make mistakes and learn from them. Misunderstandings are never used to pad out the story, and never feel cheap. Progress is never undone. It is one of the most fully realized depictions of the liminal space between two young people as they begin to fall in love - The roller coaster between bubbly feelings and crippling cringe that is first love is so difficult to portray. I've never encountered another manga that has managed to capture this specific feeling so accurately and with so much detail.

Output

84f353348a552229554fba7ba822005edcb6bca2fac8cf1735d53ae9e2915aa2e625f6d3cfa0106c8707ff0004d3ce95281b47b851b380ef91c86d2fb0e58b28

Brief Explanation of Code

My code for implementing sha512 utilizes the same structure as Professor Avi Kak’s code for implementing sha256. The differences being that the initialization bitvectors h0 – h7 are larger and there are 80 k constants instead of 64 for implementing sha512 which are also larger as it’s a 64-bit constant instead of 32-bit. The calculation for step 1 is also different as the last 128 bits must store the length of the input message instead of 64 bits and the integer multiple is calculated off a block size of 1024 bits instead of 512 bits. Step 2 is very similar as the first 16 words are calculated the same way except for sha512 they’re 64 bits instead of 32 bits and the message schedule is 80 words instead of 64. Calculating sigma0 and sigma1 is also different as the shift values are different and can be found on page 44 of lecture 15. For step 3 I just used the formulas on pages 45 and 46 of lecture 15 to calculate the hash values to store in the temporary variables. Step 4 is the exact same for sha512 and sha256 other than the modulus and size of the final bitvectors being twice the value of the modulus and size of the bitvectors for sha256.