Array Shift:

**Input:** 32-bit Integer Array I of 5 bytes {11h, 22h, 33h, 44h, 55h}

**Output:** 32-bit Byte Array R with members rotated forward one position ({55h, 11h, 22h, 33h, 44h}). The value originally at the end of the array is wrapped around to the first position of the array.

**Algorithm:**

1. Instantiate 5-byte array I (I = {1, 2, 3, 4, 5})
2. x = last value of I (5)
3. Instantiate integer S = 4 // no. of bytes in I
4. for S decrementing through 0
   1. move I[S] to the right one bit

end for // value of I should be {0,1,2,3,4} now

1. Insert value of x into the first index of R

Endian Flip:

**Input:** A 32-bit number X represented by the bytes 12h, 34h, 56h, 78h in big endian form.

**Output:** The same 4 bytes of the input represented in little endian form Y (78h, 56h, 34h, 12h).

**Algorithm:**

1. Instantiate 4-byte array X (X = {12h, 34h, 56h, 78h})
2. Instantiate 4-byte array Y (Y = {0, 0, 0, 0})
3. Instantiate integer S = 3 // highest index of X and Y
4. Instantiate integer L = 0 // loop count
5. for S decrementing through 0
   1. Y[L] = X[S]
   2. increment L