MAINLINE
sequential next
SET LMAINS-OK TO TRUE
MOVE LMAIN-MESSAGE-LEVEL TO LLOG-LEVEL OF PM-S
MOVE LMAIN-EXTRA-DG TO DG-S
MOVE LMAIN-EXTRA-TG TO TG-S
MOVE LMAIN-EXTRA-TP TO TP-S
MOVE LMAIN-EXTRA-PM TO PM-S
MOVE PM-S TO PUTMESSAGE-LS OF DG-S
MOVE PM-S TO PUTMESSAGE-LS OF TG-S
MOVE PM-S TO PUTMESSAGE-LS OF TP-S
PERFORM CHECK-INPUT
perform-exit
procedure call
MOVE WS-PUTMESSAGE TO PUTMESSAGE OF DG-S PUTMESSAGE OF TG-S PUTMESSAGE OF TP-S
CHECK-INPUT
MOVE 'Starting process' TO LTEXT OF PM-S
IF LMAIN-DATA-GET = SPACES OR LOW-VALUES
true
false
MOVE 'AESTGET' TO WS-DATA-GET
MOVE LMAIN-DATA-GET TO WS-DATA-GET
END-IF
SET LLEVEL-INF OF PM-S TO TRUE
IF LMAIN-TEXT-GET = SPACES OR LOW-VALUES
MOVE 'AESXGET' TO WS-TEXT-GET
MOVE LMAIN-DATA-GET TO WS-TEXT-GET
CALL WS-PUTMESSAGE USING PM-S
PERFORM MAIN-ACTION
MAIN-ACTION
MOVE 'Ending process' TO LTEXT OF PM-S
IF LMAIN-TEXT-PUT = SPACES OR LOW-VALUES
MOVE 'AESTPUT' TO WS-TEXT-PUT
MOVE LMAIN-TEXT-PUT TO WS-TEXT-PUT
IF LMAIN-PUT-MESSAGE = SPACES OR LOW-VALUES
MOVE LMAIN-PUT-MESSAGE TO WS-PUT-MESSAGE
MOVE 'AESMPUT' TO WS-PUT-MESSAGE
IF LMAIN-MODE = 'ECB' OR 'CBC' OR 'CFB' OR 'OFB' OR 'CTR'
MOVE LMAIN-MODE TO WS-MODE
MOVE 'ECB' TO WS-MODE
IF LMAIN-ACTION = 'CIPHER ' OR 'DECIPHER'
MOVE 'CIPHER ' TO WS-ACTION
MOVE LMAIN-ACTION TO WS-ACTION
IF LMAIN-BITS = 'L28' OR 'T92' OR 'I56'
MOVE LMAIN-BITS TO WS-BITS
MOVE 'L28' TO WS-BITS
PERFORM CLOSE-STREAMS
CLOSE-STREAMS.
GOBACK
MOVE 'Closing stream#' TO LTEXT OF PM-S
MOVE 'Closing Output' TO LTEXT OF PM-S
SET LLEVEL-INF OF PM-S TO TRUE
CALL WS-PUTMESSAGE USING PM-S
MOVE 'C' TO LXPS-CFILE OF TP-S
PERFORM CALL-TEXT-PUT
MOVE 'Closing Input' TO LTEXT OF PM-S
SET LLEVEL-INF OF PM-S TO TRUE
CALL WS-PUTMESSAGE USING PM-S
MOVE 'C' TO LXDS-CFILE OF TG-S
PERFORM CALL-TEXT-GET
MOVE 'Closing Log' TO LTEXT OF PM-S
SET LLEVEL-INF OF PM-S TO TRUE
CALL WS-PUTMESSAGE USING PM-S
MOVE 'C' TO CFILE OF PM-S
CALL WS-PUTMESSAGE USING PM-S
SET LMAINS-ERR TO TRUE
GOBACK
PERFORM CALL-TEXT-PUT
CALL-TEXT-PUT.
MOVE 32 TO LXP-TLENGTH
CALL WS-TEXTPUT USING TP-S
EVALUATE LXP-STATUS
WHEN 'ERR'
WHEN 'OK'
MOVE 'Error on TEXT-PUT' TO LTEXT OF PM-S
SET LLEVEL-MAX OF PM-S TO TRUE
CALL WS-PUTMESSAGE USING PM-S
PERFORM END-ON-ERROR
END-EVALUATE
CALL-TEXT-GET.
CONTINUE
MOVE 'Getting text,.. TO LTEXT OF PM-S
SET LLEVEL-INF OF PM-S TO TRUE
CALL WS-PUTMESSAGE USING PM-S
CALL WS-TEXTGET USING TG-S
EVALUATE LXG-STATUS
WHEN 'OK'
WHEN 'EOF'
WHEN 'ERR'
MOVE 'Error on TEXT-GET' TO LTEXT OF PM-S
SET LLEVEL-MAX OF PM-S TO TRUE
CALL WS-PUTMESSAGE USING PM-S
PERFORM END-ON-ERROR
END-ON-ERROR.
END-EVALUATE
PERFORM CALL-TEXT-GET
MOVE LDG-TEXT TO WS-CIPHER-DATA
PERFORM CALL-DATA-GET
MOVE LDG-TEXT TO WS-DECIPHER-DATA
MOVE 'Closing DATA-GET' TO LTEXT OF PM-S
SET LLEVEL-INF OF PM-S TO TRUE
CALL WS-PUTMESSAGE USING PM-S
MOVE 'C' TO LDGS-CFILE OF DG-S
PERFORM CALL-DATA-GET
CALL-DATA-GET.
CALL WS-DATAGET USING DG-S
EVALUATE LDG-STATUS
WHEN 'ERR'
WHEN 'OK'
WHEN 'EOF'
MOVE 'Error on DATAGET' TO LTEXT OF PM-S
SET LLEVEL-MAX OF PM-S TO TRUE
CALL WS-PUTMESSAGE USING PM-S
PERFORM END-ON-ERROR
END-EVALUATE
CONTINUE
CONTINUE
GET-DATA
MOVE WS-ACTION (1 : 1) TO IO-ACTION OF IOCOMM
MOVE WS-MODE TO IO-MODE OF IOCOMM
MOVE WS-BITS TO IO-BITS OF IOCOMM
MOVE PM-S TO IOPUTM
PERFORM CALL-TEXT-GET
EVALUATE WS-MODE
WHEN 'CBC'
WHEN 'CFB'
WHEN 'OFB'
WHEN 'CTR'
MOVE LMAIN-IV TO IO-TEXT OF IOCOMM
MOVE LMAIN-IV TO XOR-A
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT TO XOR-A
MOVE 'I' TO WS-C (32)
MOVE ALL '0' TO WS-CTR
END-EVALUATE
PERFORM UNTIL LXGS-EOF OR LXGS-ERR
IF LXGS-SKP
CONTINUE
EVALUATE WS-MODE ALSO WS-ACTION
WHEN 'CBC' ALSO 'DECIPHER'
WHEN 'CFB' ALSO 'DECIPHER'
WHEN 'ECB' ALSO 'DECIPHER'
WHEN 'OFB' ALSO 'DECIPHER'
WHEN 'CTR' ALSO 'CIPHER'
WHEN 'CFB' ALSO 'CIPHER'
WHEN 'ECB' ALSO 'CIPHER'
WHEN 'CTR' ALSO 'DECIPHER'
WHEN 'OFB' ALSO 'CIPHER'
WHEN 'CBC' ALSO 'CIPHER'
PERFORM CBC-DECIPHER
CBC-DECIPHER.
MOVE XOR-B TO LXP-TEXT
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
PERFORM XOR
PERFORM PAD
PERFORM ECB-DECIPHER
ECB-DECIPHER.
MOVE IO-TEXT OF IOCOMM TO XOR-B
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
PERFORM CFB-DECIPHER
CFB-DECIPHER.
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
MOVE WS-CTR TO IO-TEXT OF IOCOMM
PERFORM OFB-DECIPHER
OFB-DECIPHER.
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
PERFORM CTR-CIPHER
CTR-CIPHER.
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
PERFORM CTR-DECIPHER
CTR-DECIPHER.
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
PERFORM ECB-CIPHER
ECB-CIPHER.
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PERFORM PAD
PERFORM PAD-X
MOVE PAD-X TO IO-TEXT OF IOCOMM
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
PERFORM XOR
MOVE PAD-X TO XOR-B
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE PAD-X TO XOR-A
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PERFORM PAD
PERFORM XOR
MOVE WS-CIPHER-DATA TO IOTAB
MOVE PAD-X TO XOR-B
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE PAD-X TO XOR-A
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PERFORM PAD
PERFORM XOR
MOVE WS-CIPHER-DATA TO IOTAB
MOVE PAD-X TO XOR-B
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE PAD-X TO XOR-A
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PERFORM PAD
PERFORM XOR
MOVE WS-CIPHER-DATA TO IOTAB
MOVE PAD-X TO XOR-B
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE PAD-X TO XOR-A
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PERFORM PAD
PERFORM XOR
MOVE WS-CIPHER-DATA TO IOTAB
MOVE PAD-X TO XOR-B
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE PAD-X TO XOR-A
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PERFORM PAD
PERFORM XOR
MOVE WS-CIPHER-DATA TO IOTAB
MOVE PAD-X TO XOR-B
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE PAD-X TO XOR-A
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PERFORM PAD
PERFORM XOR
MOVE WS-CIPHER-DATA TO IOTAB
MOVE PAD-X TO XOR-B
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE PAD-X TO XOR-A
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PERFORM PAD
PERFORM XOR
MOVE WS-CIPHER-DATA TO IOTAB
MOVE PAD-X TO XOR-B
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE PAD-X TO XOR-A
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PERFORM PAD
PERFORM XOR
MOVE WS-CIPHER-DATA TO IOTAB
MOVE PAD-X TO XOR-B
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE PAD-X TO XOR-A
MOVE XOR-B TO LXP-TEXT
MOVE IO-TEXT OF IOCOMM TO LXP-TEXT
PAD.
XOR.
PERFORM HEX-ADDITION
HEX-ADD.
MOVE LXG-TEXT (1 : LXG-TLENGTH) TO PAD-X (33 - LXG-TLENGTH : LXG-TLENGTH)
MOVE ALL '0' TO PAD-X
MOVE LMAIN-IV TO XOR-A
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT TO XOR-A
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE WS-CTR TO IO-TEXT OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT OF IOCOMM TO XOR-A
MOVE WS-CTR TO IO-TEXT OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT OF IOCOMM TO XOR-A
PERFORM PAD
MOVE PAD-X TO IO-TEXT OF IOCOMM
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
PERFORM XOR
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT OF IOCOMM TO XOR-A
PERFORM HEX-ADDITION
HEX-ADD.
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT OF IOCOMM TO XOR-A
PERFORM HEX-ADDITION
HEX-ADD.
MOVE XOR-B TO IO-TEXT OF IOCOMM
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT OF IOCOMM TO XOR-A
PERFORM HEX-ADDITION
HEX-ADD.
MOVE WS-CTR TO IO-TEXT OF IOCOMM
MOVE LMAIN-KEY TO IO-KEY OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'C' TO IO-ACTION
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE IO-TEXT OF IOCOMM TO XOR-A
PERFORM HEX-ADDITION
HEX-ADD.
MOVE WS-CIPHER-DATA TO IOTAB
MOVE 'K' TO IO-ACTION OF IOCOMM
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE 'D' TO IO-ACTION OF IOCOMM
MOVE WS-DECIPHER-DATA TO IOTAB
PERFORM KEYSCHEDULE
KEYSCHEDULE
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE WS-DECIPHER-DATA TO IOTAB
MOVE 'K' TO IO-ACTION OF IOCOMM
CALL 'AESCORE' USING IOCOMM IOTAB IOPUTM
MOVE 'E' TO IO-ACTION OF IOCOMM
MOVE WS-CIPHER-DATA TO IOTAB
PERFORM VARYING XOR-I FROM 1 BY 1 UNTIL XOR-I > 16
MOVE XOR-AP (XOR-I) TO BYTE-A OF SUMM
MOVE XOR-BP (XOR-I) TO BYTE-B OF SUMM
PERFORM ADDITION
ADDITION.
MOVE BYTE-R OF SUMM TO XOR-RP (XOR-I)
MOVE WORD-A OF SUMM (1) TO W2D-A
PERFORM W2D
MOVE W2D-R TO XB-X
MOVE WORD-B OF SUMM (1) TO WORD-A
PERFORM W2D
MOVE W2D-R TO XBY
ADD 1 TO XB-X XB-Y
MOVE XB-R (XB-X XB-Y) TO WORD-B OF SUMM (1)
MOVE WORD-A OF SUMM (2) TO W2D-A
PERFORM W2D
MOVE W2D-R TO XB-X
MOVE WORD-B OF SUMM (2) TO W2D-A
PERFORM W2D
MOVE W2D-R TO XBY
ADD 1 TO XB-X XB-Y
MOVE XB-R (XB-X XB-Y) TO WORD-R OF SUMM (2)
W2D.
IF W2D-A IS NUMERIC
EVALUATE W2D-A
WHEN 'A'
WHEN 'B'
WHEN 'C'
WHEN 'D'
WHEN 'E'
WHEN 'F'
MOVE 10 TO W2D-R
MOVE 11 TO W2D-R
MOVE 12 TO W2D-R
MOVE 13 TO W2D-R
MOVE 14 TO W2D-R
MOVE 15 TO W2D-R
END-EVALUATE
MOVE W2D-A TO W2D-R
END-IF
EVALUATE WS-C (HA1)
WHEN '0'
WHEN '1'
WHEN '2'
WHEN '3'
WHEN '4'
WHEN '5'
WHEN '6'
WHEN '7'
WHEN '8'
WHEN '9'
WHEN 'A'
WHEN 'B'
WHEN 'C'
WHEN 'D'
WHEN 'E'
WHEN 'F'
MOVE '0' TO WS-C (HA1)
MOVE '1' TO WS-C (HA1)
MOVE '2' TO WS-C (HA1)
MOVE '3' TO WS-C (HA1)
MOVE '4' TO WS-C (HA1)
MOVE '5' TO WS-C (HA1)
MOVE '6' TO WS-C (HA1)
MOVE '7' TO WS-C (HA1)
MOVE '8' TO WS-C (HA1)
MOVE '9' TO WS-C (HA1)
MOVE 'A' TO WS-C (HA1)
MOVE 'B' TO WS-C (HA1)
MOVE 'C' TO WS-C (HA1)
MOVE 'D' TO WS-C (HA1)
MOVE 'E' TO WS-C (HA1)
MOVE 'F' TO WS-C (HA1)
SET SW-CARRY-F TO TRUE
END-EVALUATE
END-IF
test