

07/05/73

ASSEMBLER LISTING OF EVOS TERMINAL SYSTEM

```

991      B      IFT4B
992      IFT4C  L      REG4,IFNUM      STORE NEW PATTERN
993      LA      REG4,IFLIST(REG4)
994      O      1,4(REG4)      ADD OLD TO NEW
995      ST      1,4(REG4)      STORE NEW
996      BR      14
997      IFA1   NI      ACTIVE,X'FC'  MAKE SJKE ZERO (AND FUNCTION)
998      OI      ACTIVE,X'02'  SET BIT
999      BR      14
1000      IF01  NI      ACTIVE,X'FC'  OR FUNCTION
1001      OI      ACTIVE,X'01'
1002      BR      14
1003      IFX1  NI      ACTIVE,X'FC'  XOR FUNCITON
1004      OI      ACTIVE,X'03'
1005      BR      14
1006      IFTGO *      ST      14,SAVE2
1007      *
1008      XI      TINTFG,X'80'  FILL-IN QTAB W/ TESTS
1009      DOIO   SYSPRINT,IFNMSG
1010      L      14,SAVE2
1011      B      TYGD
1012      *
1013      IFN2   L      REG4,PARM      NAME=XXX
1014      BVALUE B,REG4,REG4
1015      STH    REG4,ACTIVE+4
1016      BR      14
1017      EJECT
1018      *
1019      TP2   EQU      TALLY COMMANDS
1020      TF2   SAVER   SAVE=SAVE2
1021      TF2A  TM      TINTFG,X'10'
1022      BZ    TF2B
1023      BAL   14,TFYFILL
1024      TF2B L      REG5,P1YPE
1025      A      REG5,=F'1'
1026      ST      REG5,P1YPE
1027      XR      REG4,REG4
1028      M      REG4,=F'4'  REG5=A(1TFCHK)+4*C(P1YPE)
1029      LA     REG5,TFCHK-4(REG5)  REG5=A(NEXT FORM NAME)
1030      L      REG4,PARM
1031      MVC   0(4,REG5),1(REG4)  MOVE IN FORM NAME
1032      TM    QFS+1,X'80'  TURN ON 'LAST QUES OF CARD', FLAG
1033      BO    TF2C      BUT NOT IF 1ST QUES
1034      L      REG2,REG2STRE
1035      A      REG2,=F'-16'
1036      OI    QFS+1,X'01'
1037      ST    REG2,REG2STRE
1038      TF2C  L      13,SAVE2+4
1039      LM    14,12,12(13)
1040      L      REG2,REG2STRE
1041      BR    14
1042      *
1043      *      NOW INDEX=XX
1044      TQ2   TM      TINTFG,X'10'
1045      ST    14,B(13)

```

```

1046          BZ      TQ2A
1047          BAL     14, TYFILL
1048          OI      TINTFG, X'10'          SET TALLY FILL FLAG
1049 TQ2A      MVC     TONUM, =F'1'          ONLY ONE QUES GIVEN
1050          L        REG3, PARM
1051          BVALUE  2, REG3, REG3
1052          STH     REG3, TQUES
1053          L        14, 8(13)
1054          BR      14
1055          *
1056          TQ3      TM      TINTFG, X'10'          QUES=(XX, XX, XX)
1057          ST      14, 8(13)
1058          BZ      TQ3A
1059          BAL     14, TYFILL
1060          OI      TINTFG, X'10'          SET TALLY FILL FLAG
1061 TQ3A      L        REG5, TONUM
1062          XR      REG4, REG4
1063          A        REG5, =F'1'
1064          ST      REG5, TONUM          NEW NUM OF QUES
1065          M        REG4, =F'2'          CALC OFFSET IN TQUES
1066          L        REG3, PARM
1067          BVALUE  2, REG3, REG3
1068          STH     REG3, TQUES-2(REG5)    STORE NEW NUMBER
1069          L        14, 8(13)
1070          BR      14
1071          *
1072          TQ4      TM      TINTFG, X'10'
1073          ST      14, 8(13)
1074          BZ      TQ4AA
1075          BAL     14, TYFILL
1076          OI      TINTFG, X'10'          SET TALLY FILL FLAG
1077 TQ4AA     L        CNT, PARM+4          HI VALUE
1078          BVALUE  2, CNT, CNT
1079          L        REG3, PARM          PARAMS LEN2
1080          BVALUE  2, REG3, REG3          LOW VALUE
1081          SR      CNT, REG3          DIFFERENCE
1082          LTR     CNT, CNT
1083          BNP     TQ4AB
1084          L        REG5, TONUM          CALC SUBSCRIPT TQUES
1085          XR      REG4, REG4
1086          LR      1, REG5
1087          AR      1, CNT
1088          ST      1, TONUM          STORE NEW NUM OF QUES
1089          M        REG4, =F'2'
1090 TQ4A      STH     REG3, TQUES(REG5)
1091          A        REG3, =F'1'
1092          A        REG5, =F'2'
1093          BCT     CNT, TQ4A
1094 TQ4AB     L        14, 8(13)
1095          BR      14
1096          *
1097          TQ2     L        REG3, PARM
1098          BVALUE  1, REG3, REG3          PARAM IS LEN1
1099          LNR     REG3, REG3
1100          A        REG3, =F'8'

```

```

1101 L REG4,=F'1'
1102 SLL REG4,0(REG3) TURN REQUESTED BIT ON IN REG4
1103 EX REG4,T0201 TURN IT ON IN ACTIVE
1104 QI ACTIVE+1,X'20' TURN FORMATING ON
1105 BB 14
1106 T0201 OI ACTIVE+2,X'00'
1107 * HOLD
1108 TH1 OI ACTIVE,X'08'
1109 BR 14
1110 * GO
1111 TG1 NI ACTIVE,B'11110111'
1112 BR 14
1113 * FINE
1114 TF1 NI ACTIVE,B'11110111'
1115 OI INTFG,X'20'
1116 BR 14
1117 * TAB=10
1118 TT3 EQU *
1119 TT2 L REG3,PARM REG3 HAS TAB LEN
1120 BVALUE 2,REG3,REG3 LEN2 OF PARAM
1121 TT2A XR REG5,REG5 TABSS SETING,LEN(H,H)
1122 LR REG4,REG3 TAB SETTING=REG4
1123 * **** MARGIN SETTING FOLLOWS I*****
1124 A REG4,=F'12' MARGIN OF 12
1125 TT2B STH REG4,TABSS(REG5)
1126 STH REG3,TABSS+2(REG5)
1127 AR REG4,REG3
1128 A REG4,TABGUARD GUARD BAND
1129 A REG5,=F'4'
1130 CL REG5,=F'64'
1131 BL TT2B COULD PRINT TAB LINE-ECH
1132 BR 14
1133 *
1134 EVQTT1 EQU *
1135 TT1 SAVE (14,12)
1136 L REG3,=F'8' REG3=DEFAULT TAB LEN.
1137 BAL TT2A
1138 RETURN (14,12)
1139 *
1140 TT4 L REG5,PARMH4 CHANGE SPECIFIED TABS TAB=(01-12)
1141 BVALUE 2,REG5,REG5 TAB TO BE CHANGED
1142 XR REG4,REG4 LEN 2 OF PARAM
1143 M REG4,=F'4' CALC SUBSCRIPT FOR TABSS
1144 L REG4,PARM NEW LEN
1145 BVALUE 2,REG4,REG4 LEN 2 OF PARAM
1146 STH REG4,TABSS+2(REG5) STORE NEW LEN
1147 TT4A LH REG3,TABSS(REG5) OLD SETTING
1148 AH REG3,TABSS+2(REG5) LEN
1149 A REG3,TABGUARD GUARD BAND
1150 A REG5,=F'4' REG3=NEW NEXT SETTING
1151 CL REG5,=F'60'
1152 BH TT4B
1153 STH REG3,TABSS(REG5) STORE NEW SETTING
1154 B TT4A
1155 TT4B BR 14

```

```

1153 *                                     SPECIAL=(1) NR FLAG
1157 TS3  L   REG3,PARM
1158     CLI 1(REG3),C'I'
1159     BE  TS3A
1160     OI ACTIVE,X'10' SAVE
1161     BR 14
1162 TS3A NI ACTIVE,B'11101111' IGNORE
1163     BR 14
1164 *                                     SPECIAL=(1-1) (99-WIERD)
1165 TS4  L   REG3,PARM+4 PROCESS'99
1166     CLI 1(REG3),C'I'
1167     BE  TS4A
1168     OI ACTIVE,X'40' SAVE
1169     B   TS4A
1170 TS4A NI ACTIVE,B'10111111' IGNORE
1171 TS4B L   REG3,PARM PROCESS WIERD
1172     CLI 1(REG3),C'I'
1173     BE  TS4C
1174     OI ACTIVE,X'80'
1175     BR 14
1176 TS4C NI ACTIVE,X'7F'
1177     BR 14
1178 TD1  BR 14 USE PDUMP INSTEAD
1179 *                                     GET ROOM IN QORKA
1180 * GETSTORE SAVE (2,14)
1181 *                                     DO HALF WORK ALIGNMENT
1182     L   REG5,QSTORE
1183     LR  REG3,REG5
1184     XR  REG4,REG4
1185     D   REG4,=F'2'
1186     LTR REG4,REG4
1187     BZ  GSHF2
1188     A   REG3,=F'1'
1189 GSHF2 EQU *
1190     LTR REG5,1
1191     BM  GSFW
1192     A   REG5,QSTORE
1193     XR  15,15
1194     CL  REG5,SYSLN
1195     DH  GSTERM
1196     BL  GSGOOD
1197     DUIO SYSPRINT,GSWARN
1198     L   15,=F'-4'
1199 GSGOOD L   1,QSTORE ADR RETURNED
1200     ST  REG5,QSTORE FOR NEXT REQUEST-UPDATE QSTORE
1201     RETURN (2,14)
1202 GSFW  LPR  REG5,1
1203     A   REG5,QSTORE
1204     XR  REG4,REG4
1205     D   REG4,=F'4'
1206     LTR REG4,REG4 ZERO IF ON BOUNDRY
1207     BZ  GSGOOD
1208     A   REG5,QSTORE
1209     L   1,QSTORE
1210     A   1,=F'2'

```

07/05/73

ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

```

1211          ST      REG5, QSTORE
1212          RETURN  (2, 14)
1213          GSTERM  DOIO  SYSPRINT, GSTERMG
1214          L        15, =F'16'
1215          BAL     14, TFI
1216          LM      2, 14, 12(13)
1217          BR      12, 14
1218          GSTERMG  DC     CL80, WORKA, STORAGE EXCEEDED-ISSUED FINISH
1219          GSWARN  *      DC     CL80, WORKA, STORAGE FILLED-WARNING
1220          *
1221          *
1222          TYFILL  SAVER  (SAVE=SAVE3      TALLY FILL-IN ROUTINE
1223          L        REG2, REG2STRE
1224          XI      TINTFG, X'01'
1225          *
1226          DO      CL(N)      TURN-OFF FILL FLAG
1227          LH      REG3, Q03      REG3=OFST(BTL)
1228          LA      REG5, WORKA+2(REG3)  N
1229          XR      REG5, =F'3'
1230          M      REG4, RBG4
1231          L      REG4, =F'2'      CALL LEN FOR GETSTORE
1232          LR      1, REG5
1233          BAL     14, GETSTORE
1234          LTR     15, 15
1235          BZ      TYFILL1
1236          TYSPT  DOIO  SYSPRINT, TYFILLM1
1237          BAL     14, TFI
1238          L        15, SAVE3+4
1239          RETURN  (14, 12)
1240          TYFILLM1 DC     CL80, RAN OUT OF WORK SPACE-NO CHANCE FOR RUN
1241          TYFLMVC MVC     0(0, 1), 0(COMBASE)
1242          TYFLX1  XC     0(0, 1), 0(1)      ZERO AREA
1243          TYFILL1 EX     REG5, TYFLX1      REG5 HAS LENGTH REQUESTED FOR CL(N)
1244          LH      REG3, Q03      CALC OFFSET OF CL FROM BTL
1245          LA      REG4, WORKA(REG3)
1246          LR      COMBASE, REG4
1247          SR      REG4, 1
1248          STH     REG4, WORKA+4(REG3)  STORE OFFSET OF CL IN BTL
1249          AR      REG4, REG5      REG4=TOTAL LENGTH(IMAGE)
1250          L        CNT, TQNUM      DOWN COUNTER
1251          LA      REG3, TQUES      A(QUES NUM TO BE FILLED)
1252          LA      REG5, WORKA
1253          *
1254          *      HAVE TO USE 2 BASE REG5 FOR OTHER THINGS
1255          *      THESE AREN'T NEED NOW & ARE RESTORED WITH RETURN
1256          *      COMBASE=A(START OF IMAGE)
1257          *      DCBASE=LEN OF IMAGE
1258          LR      DCBASE, REG4
1259          A        DCBASE, =F'-1'
1260          LNR     REG4, REG4      FULL WORK GETSTORE=-L(IMAGE)
1261          LNR     REG5, REG5      REG5=-A(WORKA)
1262          *      START ACTJAL FILL OPERATION
1263          TYFLA  EQU     *
1264          *      COPY PROTO-TYPE FROM ACTIVE
1265          MVC     QNUM, 0(REG3)
1266          MVC     QFS+2(2), ACTIVE+2
1267          MVC     Q01(2), ACTIVE+4

```

07/05/73

ASSEMBLER LISTING OF EVOS TERMINAL SYSTEM

PAGE 24

```

1266 *          GENERATE IMAGE FOR ACTIVE QUES.
1267          LIR          REG4
1268          BAL          14,GETSTORE
1269          LTR          15,15
1270          BNE          TYSPT
1271          EX           DCBASE,TYFLMVC
1272 *          MVC          IMAGE AT COMBASE W/ LEN OF DCBASE TO
1273 *          ADR RETURNED IN REG1
1274          AR           1,REG5          CALC OFFSET OF IMAGE IN WORKA
1275          STH          1,003          STORE OFFSET
1276 *          FINISHED GENERATION-NEXT QUES NOW
1277          A           REG3,=F'2'
1278          A           REG2,=F'16'
1279          BCT          CNT,TYFLA
1280 TYFLRT      ST          REG2,REG2STRE
1281          XC          003,003
1282          XC          TONUM,TONUM
1283          XC          ACTIVE+2(2),ACTIVE+2
1284          L           13,SAVE3+4
1285          RETURN      (14,12)
1286 *          GO SEQUENCE
1287 TYGO        SAVER      SAVE=SAVE2
1288          TM          TINTFG,X'20'    CHK FINE FLAG
1289          BZ          TYGOB
1290          NI          ACTIVE,X'F7'    TURN HOLD FLAG OFF
1291          XI          TINTFG,X'20'    TURN FINE FLAG OFF
1292          L           13,SAVE2+4
1293          RETURN      (14,12)    END
1294 *
1295 TYGOB       TM          ACTIVE,X'08'  CHK HOLD FLAG
1296          BZ          TYGOA
1297          L           13,SAVE2+4
1298          RETURN      (14,12)
1299 *
1300 TYGOA       TM          TINTFG,X'10'  CHK TALLY FILL FLAG
1301          BZ          TYGDC
1302          BAL          14,TYFILL
1303 TYGOC       L           REG2,REG2STRE
1304          A           REG2,=F'-16'
1305          OI          QFS+1,X'03'    SET 'LAST QUES OF ALL' FLAG
1306          CALL        EVQEOQR
1307          CALL        EVQTYOUT
1308          L           13,SAVE2+4
1309          RETURN      (14,12)
1310          TITLE      'BUILDING WORK LISTS FOR TRANSLATORS'
1311 *          BTL-INITIALIZING SEQUENCE
1312 BTLINIT     SAVER      SAVE=SAVE3
1313          OI          TINTFG,X'10'    SET TALLY FILL FLAG
1314          L           1,=F'-10'      LEN & FULL WORK BOUNDARY
1315          BAL          14,GETSTORE
1316          LTR          15,15
1317          BNZ         BTLRTN
1318          LA          REG3,WORKA
1319          SR          1,REG3
1320          STH          1,003

```

```

1321 BTLRTN L 13,SAVE3+4 FILL-IN
1322 LM 14,12,12(3)
1323 LH 1,003
1324 BR 14
1325 *
1326 * KNOW INITIAL CALL BY Q03=H'0'
1327 STYPEMVC MVC, 0(0,1),1(REG4) MVC, FROM C(REG4)+1 TO C(1)
1328 WKLTRTN L 13,SAVE2+4
1329 RETURN 14,12
1330 *
1331 TC3 L CHAR TRAN WORK LIST
1332 CLC REG2,REG2STRE
1333 BE Q03,=F'0' EXTEND CHAR WORK LIST
1334 SAVER SAVE=SAVE2
1335 L REG2,REG2STRE
1336 * BTL(N=N+1)
1337 LH REG3,Q03 0(BTL) IN WORKA
1338 LH REG4,WORKA+2(REG3) LOAD N
1339 A REG4,=F'1'
1340 STH REG4,WORKA+2(REG3) N=N+1
1341 * S-TYPE(YN=PARM)
1342 TC3A LH REG5,WORKA(REG3) LOAD L
1343 LR 1,REG5
1344 BAL 14,GETSTORE
1345 LTR 15,15
1346 BP WKLTRTN
1347 L REG4,PARM
1348 BX REG5,STYPEMVC STORE PARM
1349 B WKLTRTN
1350 * INITIAL CHAR WL
1351 TC2 SAVER SAVE=SAVE2
1352 L REG2,REG2STRE
1353 * HVI ACTIVE+3,X'80' SET TRANS. BIT ON
1354 BTL(L=PARML-1,N=1)
1355 BAL 14,BTLINIT
1356 LR REG3,1
1357 L REG4,PARML
1358 A REG4,=F'-1'
1359 STH REG4,WORKA(REG3) L=PARML-1
1360 L REG4,=F'1'
1361 STH REG4,WORKA+2(REG3) N=1
1362 B TC3A S-TYPE(Y1=PARM)
1363 * MATCH WORK LIST
1364 * SIMILAR TO CHAR EXCEPT VARIABLE LENGTH
1365 TM3 CLC Q03,=F'0' EXTEND MATCH WL
1366 RE TM2
1367 SAVER SAVE=SAVE2
1368 L REG2,REG2STRE
1369 * BTL(L=MAX(PARML-1,L),N=N+1)
1370 LH REG3,Q03 0(BTL) IN WORKA
1371 LH REG4,WORKA+2(REG3) LOAD N
1372 A REG4,=F'1'
1373 STH REG4,WORKA+2(REG3) N=N+1
1374 LH REG4,WORKA(REG3) LOAD L
1375 L REG5,PARML

```

07/05/73

ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

```

1376 A REG5,=F'-1' REG5=PARML-1
1377 CLR REG4,REG5
1378 BH TM3B
1379 LR REG4,REG5
1380 TH3B STH REG4,WORKA(REG3) REG4=MAX(PARML-1,L)
1381 * S-TYPE(YN=(PARML-1,PARM))
1382 TM3A LR L,REG5
1383 A L,=F'2' HALFWORD FOR STORING LENGTH
1384 BAL L,GETSTORE
1385 LTR L,=F'5'
1386 BP WKLTRN
1387 STH REG5,0(L) STORE LEN. OF POSS.
1388 A L,=F'2'
1389 L REG4,PARM
1390 EX REG5,STYEMVC STORE PARM
1391 B WKLTRN
1392 * INITIAL MATCH WL
1393 TM2 SAVER SAVE=SAVE2
1394 L REG2,REG2STRE
1395 MVI ACTIVE+3,X'10' SET TRANS BIT ON
1396 * BTL(L=PARML-1,N=1)
1397 BAL L,=F'4'
1398 LR REG3,1
1399 L REG5,PARML
1400 A REG5,=F'-1'
1401 STH REG5,WORKA(REG3) L=PARML-1
1402 L REG4,=F'1'
1403 STH REG4,WORKA+2(REG3) N=1
1404 B TM3A S-TYPE(Y1=(PARML-1,PARM))
1405 * NUMERIC MULTIPLE CHOICE WL
1406 TA3 EQU * AMC INITIAL
1407 TN2 SAVER SAVE=SAVE2 NMC
1408 MVI ACTIVE+3,X'140'
1409 L REG2,REG2STRE
1410 * BTL(L=0,N=V(PARM))
1411 BAL L,=F'4'
1412 LR REG3,1
1413 L REG4,PARM
1414 BVALUE 2,REG4,REG4 PARAM HAS LEN2
1415 STH REG4,WORKA+2(REG3) N=V(PARM)
1416 XR REG4,REG4
1417 STH REG4,WORKA(REG3) L=0
1418 B WKLTRN
1419 * ALPHADETIC MULTIPLE CHOICE
1420 * USE TN2 FOR INITIAL
1421 TA4 SAVER SAVE=SAVE2 EXTEND
1422 MVI ACTIVE+3,X'20'
1423 L REG2,REG2STRE
1424 * BTL(L=L+1,N=N+H2)
1425 LH REG3,003 0(BTL) IN WORKA
1426 LH REG4,WORKA(REG3) LOAD L
1427 A REG4,=F'1'
1428 STH REG4,WORKA(REG3) L=L+1
1429 * N-TYPE(H1=PARM+4,H2=PARM)
1430 L L,=F'4'

```

345

07/05/73

ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

PAGE 27

```

1431 BAL 14,GETSTORE
1432 LTR 15,15
1433 BP WKLTRTN
1434 LR REG3,1
1435 L REG4,PARM+4
1436 BVALUE 2,REG4,REG4 PARM HAS LEN2
1437 STH REG4,WORKA(REG3) H1=PARM+4
1438 L REG4,PARM
1439 BVALUE 1,REG4,REG4 PARM HAS LEN1
1440 STH REG4,WORKA+2(REG3) H2=PARM=REG4
1441 LH REG3,Q03 REG3=0( TL)
1442 AH REG4,WORKA+2(REG3)
1443 STH REG4,WORKA+2(REG3) N=N+H2
1444 B WKLTRTN
1445 * RANGE WURK LIST
1446 * 2ND HALFWORD IS H'0' FOR FORM3
1447 TR3 LA REG4,=F'0'
1448 ST REG4,PARM PARM=F'0'
1449 L REG4,=F'2'
1450 ST REG4,PARML
1451 TR4 SAVER SAVE=SOVE2
1452 MVI ACTIVE+3,X'08'
1453 L REG2,REG2STRE
1454 CLC Q03,=F'0'
1455 BE TR4XTND
1456 *
1457 BTL(L=0,N=1) INITIAL
1458 BAL 14,BTLINLT
1459 LR REG3,1
1460 L REG4,=F'1'
1461 STH REG4,WORKA+2(REG3) N=1
1462 XR REG4,REG4
1463 STH REG4,WORKA(REG3) L=0
1464 * N-TYPE(H1=PARM+4,H2=PARM)
1465 TR4A L 1,=F'4'
1466 BAL 14,GETSTORE
1467 LTR 15,15
1468 BP WKLTRTN
1469 LR REG3,1
1470 L REG4,PARM+4
1471 BVALUE 5,REG4,REG4 PARM HAS LENS
1472 STH REG4,WORKA(REG3) H1=PARM+4
1473 L REG4,PARM
1474 BVALUE 5,REG4,REG4 PARM HAS LENS
1475 STH REG4,WORKA+2(REG3) H2=PARM
1476 B WKLTRTN
1477 * TR4XTND BTL(N=N+1) EXTEND
1478 LH REG3,Q03
1479 L REG4,WORKA+2(REG3) LOAD N
1480 A REG2,=F'1'
1481 STH REG4,WORKA+2(REG3) N=N+1
1482 B TR4A
1483 TX1 BR 14
1484 TY1 BR 14
1485 TZ1 BR 14
LORG

```

07/05/73

ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

PAGE 28

```

1486 EVQEQQR BEGIN BASE=BASE
1487 SAVED SAVE=SAVEQ
1488 CLC WORKIN,WORKOUT
1489 BE EQORFLNG
1490 * OPEN (WORKIN,WORKOUT,OUTPUT)
1491 LA 1,WORKIN
1492 MVI WORKOUT,X'BF'
1493 * SET OPTION BYTE ON WORKOUT (OUTPUT, LAST FILE IN OPEN)
1494 SVC
1495 MVI WORKOUT,X'00'
1496 * OPEN (MASPRINT,OUTPUT)
1497 LA 1,MASPRINT
1498 MVI MASPRINT,X'BF'
1499 SVC
1500 MVI MASPRINT,X'00'
1501 NI ACTIVE+1,X'F0'
1502 XQ ACTIVE+2(2),ACTIVE+2 9ERO IT SO CAN
1503 * TOTAL STATE
1504 LA REG2,UTAR
1505 A REG2,=F'16'
1506 QN04 IA REG2,=F'16'
1507 OD ACTIVE+2(2),QFS+2
1508 TM QFS+1,X'01'
1509 BZ QN04
1510 EOF WORKIN,QN02
1511 TM ACTIVE+2,B'01111100'
1512 BM QL2
1513 * LISTINGS ONLY
1514 TM ACTIVE+1,X'20'
1515 BO QL2
1516 * DRAW LISTING
1517 QN01 DDIO WORKIN,OUT
1518 DDIO WORKOUT,OUT
1519 B QN01
1520 SAVEQ OS 18F
1521 * CLOSE FILES
1522 QN02 CLC MASPRINT,SCANF
1523 BU QN02A
1524 LA 1,MASPRINT
1525 MVC MASPRINT,X'80'
1526 SVC 20
1527 QN02A CLC WORKOUT,SCANF CHK SO DONT CLOSE SYSPRINT/MASPRINT
1528 BL QN03
1529 LA 1,WORKIN
1530 MVC WORKOUT,X'80'
1531 SVC 20 CLOSE (WORKIN,WORKOUT)
1532 MVI WORKOUT,X'00'
1533 L 13,SAVEQ+4
1534 RETURN (14,12)
1535 QN03 MVI WORKIN,X'80'
1536 SVC 20 CLOSE (WORKIN) ONLY
1537 MVI WORKIN,X'00'
1538 L 13,SAVEQ+4
1539 RETURN (14,12)
1540 EQORFLNG DDIO SYSPRINT,EQFLMG

```

347

07/05/73

ASSEMBLER LISTING OF EVOS TERMINAL SYSTEM

```

1541 L RETURN 13,SAVEQ+4
1542 RETURN (14,12)
1543 EQFLMG DC CL80 THE SAME FILE IS ASSIGNED TO INPUT + OUTPUT UNITS
1544 * ANALYSIS OVERHEAD SECTION
1545 * STORE OFFSET OF EQQ IN QLOC
1546 * REG2 IS BASE FOR QTAB
1547 * LOCATE EQQ'S GEOL'S (REPLACE EOL'S WITH EQQ'S)
1548 CRDCHKCP DS CL8 FORM/PAGE OF CARD TO COMPARE WITH
1549 CROSTRE DS CL80 PFSUDD-BUFFER
1550 QL2 EQU NEW RESP. NO.
1551 *
1552 LA REG2=A(QLOC)
1553 * CHAR,CARDIN+10
1554 XR CNT,CNT A(CHAR CHECKED)
1555 LR REG3,CNT COUNT QUES. NO.
1556 L REG5,=F'1' COUNT LEN.
1557 LR REG4,REG3
1558 LA REG2,QTAB OFFSET OF CHAR
1559 MVI CARDIN,X'FF' A(QTAB)
1560 NI OPS+1,X'10111111' ZERO EOP FLAG
1561 NI ACTIVE+1,X'FC' ZERO LEN FLAG
1562 *
1563 QLOTS DOTO WORKIN,CROSTRE
1564 *
1565 GLI CARDIN,X'FF' FIRST CARD WITH RESP. NO.
1566 BNE QLOTS3
1567 MVC CARDIN(11),CROSTRE
1568 B QLOTS
1569 QLOTS3 CLC CROSTRE+3(5),CARDIN+3
1570 BE QLOTS SAME RESP. NU. & WANTED
1571 DOIO SYSPRINT,NGFILEMG
1572 DOIO SYSPRINT,TFCHK
1573 DOIO SYSPRINT,CROSTRE
1574 DOIO SYSPRINT,CARDIN
1575 B QNO2
1576 NGFILEMG DC C INPUT FILE DOES NOT CONTAIN THE CARDS REQUESTED
1577 DC CL34 BY THE TALLY COMMAND
1578 * CARD CHECK SECTION
1579 QLOTS L CNT,PTYPE
1580 LA REG4,TFCHK
1581 MVC CRDCHKCP(4),CROSTRE FORM
1582 MVC CRDCHKCP+4(4),CROSTRE+8
1583 QLOTS3 CLC O(8;REG4),CRDCHKCP
1584 BE QLOTS4
1585 A REG4,=F'8'
1586 OCT CNT,QLOTS3 NEXT ONE
1587 B QLOTS
1588 QLOTS4 MVC 1(89,CHAR),CROSTRE+11
1589 QLS3 AR CHAR,REG5
1590 AR REG4,REG5
1591 AR REG3,REG5
1592 CLI O(CHAR),C'=' CHK FOR EOP
1593 BNE QLS3
1594 TH ACTIVE+1,X'01'
1595 BZ QLS3

```

```

1596      OI      'ACTIVE+1,X'02'
1597      OI      'QFS+1,X'40'      SET EOP FLAG
1598      B       'Q03
1599      QL3A   CLI      'O(CHAR),C'      CHK FOR EQO
1600      BE      'QL4
1601      CLI      'O(CHAR),C'      CHK FOR EOL: (0,2)
1602      BNE     'Q03
1603      MVI     'O(CHAR),C'      REPLACE EOL WITH EQO (L,2)
1604      XR      'CNT,CNT      FINISH CARD
1605      TM      'ACTIVE+1,X'01'
1606      BO      'Q07
1607      STH     'REG3,2(REG2)      STORE LEN
1608      XR      'REG3,REG3
1609      NI      'ACTIVE+1,X'FE'
1610      QL7   TM      '5(REG2),X'03'
1611      BO      'Q00      LAST CARD
1612      BM      'Q00TS      NEXT CARD
1613      LA      'REG2,=F'16'      ADVANCE TO END OF THIS CARD
1614      NI      'QFS+1,X'BF'      ZERO EOP FLAG
1615      B       'Q07
1616      *
1617      *
1618      QL4   TM      'ACTIVE+1,X'01'
1619      BO      'Q06
1620      STH     'REG3,2(REG2)      STORE LEN
1621      XR      'REG3,REG3
1622      QL6   XI      'ACTIVE+1,X'01'      ZERO LEN FLAG
1623      AR      'CNT,REG5      NEXT QUES
1624      TH      '5(REG2),X'03'
1625      BZ      'Q08      MORE QUES-THIS CARD
1626      BO      'Q08      LAST QUES OF LAST CARD
1627      XR      'CNT,CNT      RESTART CNT FOR NEXT CARD
1628      B       'Q03      MORE CARDS
1629      QL8   A       'REG2,=F'16'
1630      NI      'QFS+1,X'BF'      ZERO EOP FLAG
1631      CH      'CNT,8(REG2)      BC IF NOT ACTIVE QUES
1632      BL      'Q03
1633      STH     'REG4,0(REG2)      YES-STORE OFFSET
1634      OI      'ACTIVE+1,X'01'      TURN ON LEN FLAG
1635      B       'Q03
1636      QBRLIST OC      'V(EVQST,EVQNM, EVQANC, EVQMATCH, EVORNGE)
1637      DC      'V(EVQUSERX, EVQUSERY, EVQUSERZ)
1638      QL0   TM      'ACTIVE+2,B'01111100'
1639      BM      'QF2
1640      *
1641      *
1642      LA      '14,Q00TS      FORMATED LISTING ONLY
1643      B       'QLLISTA
1644      QBTH  TM      'ACTIVE+3,X'00'
1645      *
1646      QF2   LI      'REG3,=F'11'      TRANSLATION BRANCH ROUTINE
1647      XR      'REG4,REG4
1648      QF3   EX      'REG3,QBTH
1649      BZ      'QF4
1650      *
1651      *
1652      BR QBRLIST(REG4)
1653      BALR   '14,BR      BR TO ROUTINE
1654      *
1655      *
1656      *
1657      *
1658      *
1659      *
1660      *
1661      *
1662      *
1663      *
1664      *
1665      *
1666      *
1667      *
1668      *
1669      *
1670      *
1671      *
1672      *
1673      *
1674      *
1675      *
1676      *
1677      *
1678      *
1679      *
1680      *
1681      *
1682      *
1683      *
1684      *
1685      *
1686      *
1687      *
1688      *
1689      *
1690      *
1691      *
1692      *
1693      *
1694      *
1695      *
1696      *
1697      *
1698      *
1699      *
1700      *
1701      *
1702      *
1703      *
1704      *
1705      *
1706      *
1707      *
1708      *
1709      *
1710      *
1711      *
1712      *
1713      *
1714      *
1715      *
1716      *
1717      *
1718      *
1719      *
1720      *
1721      *
1722      *
1723      *
1724      *
1725      *
1726      *
1727      *
1728      *
1729      *
1730      *
1731      *
1732      *
1733      *
1734      *
1735      *
1736      *
1737      *
1738      *
1739      *
1740      *
1741      *
1742      *
1743      *
1744      *
1745      *
1746      *
1747      *
1748      *
1749      *
1750      *
1751      *
1752      *
1753      *
1754      *
1755      *
1756      *
1757      *
1758      *
1759      *
1760      *
1761      *
1762      *
1763      *
1764      *
1765      *
1766      *
1767      *
1768      *
1769      *
1770      *
1771      *
1772      *
1773      *
1774      *
1775      *
1776      *
1777      *
1778      *
1779      *
1780      *
1781      *
1782      *
1783      *
1784      *
1785      *
1786      *
1787      *
1788      *
1789      *
1790      *
1791      *
1792      *
1793      *
1794      *
1795      *
1796      *
1797      *
1798      *
1799      *
1800      *
1801      *
1802      *
1803      *
1804      *
1805      *
1806      *
1807      *
1808      *
1809      *
1810      *
1811      *
1812      *
1813      *
1814      *
1815      *
1816      *
1817      *
1818      *
1819      *
1820      *
1821      *
1822      *
1823      *
1824      *
1825      *
1826      *
1827      *
1828      *
1829      *
1830      *
1831      *
1832      *
1833      *
1834      *
1835      *
1836      *
1837      *
1838      *
1839      *
1840      *
1841      *
1842      *
1843      *
1844      *
1845      *
1846      *
1847      *
1848      *
1849      *
1850      *
1851      *
1852      *
1853      *
1854      *
1855      *
1856      *
1857      *
1858      *
1859      *
1860      *
1861      *
1862      *
1863      *
1864      *
1865      *
1866      *
1867      *
1868      *
1869      *
1870      *
1871      *
1872      *
1873      *
1874      *
1875      *
1876      *
1877      *
1878      *
1879      *
1880      *
1881      *
1882      *
1883      *
1884      *
1885      *
1886      *
1887      *
1888      *
1889      *
1890      *
1891      *
1892      *
1893      *
1894      *
1895      *
1896      *
1897      *
1898      *
1899      *
1900      *
1901      *
1902      *
1903      *
1904      *
1905      *
1906      *
1907      *
1908      *
1909      *
1910      *
1911      *
1912      *
1913      *
1914      *
1915      *
1916      *
1917      *
1918      *
1919      *
1920      *
1921      *
1922      *
1923      *
1924      *
1925      *
1926      *
1927      *
1928      *
1929      *
1930      *
1931      *
1932      *
1933      *
1934      *
1935      *
1936      *
1937      *
1938      *
1939      *
1940      *
1941      *
1942      *
1943      *
1944      *
1945      *
1946      *
1947      *
1948      *
1949      *
1950      *
1951      *
1952      *
1953      *
1954      *
1955      *
1956      *
1957      *
1958      *
1959      *
1960      *
1961      *
1962      *
1963      *
1964      *
1965      *
1966      *
1967      *
1968      *
1969      *
1970      *
1971      *
1972      *
1973      *
1974      *
1975      *
1976      *
1977      *
1978      *
1979      *
1980      *
1981      *
1982      *
1983      *
1984      *
1985      *
1986      *
1987      *
1988      *
1989      *
1990      *
1991      *
1992      *
1993      *
1994      *
1995      *
1996      *
1997      *
1998      *
1999      *
2000      *

```

07/05/73

***** ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

```

1651 QF4 A REG4,=F'4'
1652 SLA REG3,1(0)
1653 CL REG4,=F'512'
1654 BL QF3
1655 *
1656 TM OUTPUT ROUTINES
1657 BZ ACTIVE+2,B'01010100' SEE IF ANY COND. NEEDED
1658 QF5 BAL 14,QLCOND B COND
1659 STM ACTIVE+2,X'AD'
1660 BZ QF6
1661 DAL 14,QLLIST B LIST
1662 QF6 TM ACTIVE+2,X'30'
1663 BZ QF7
1664 BAL 14,QLTALLY B TALLY
1665 QF7 TM ACTIVE+2,X'0A'
1666 BZ QLE
1667 BAL 14,QLBINARY B LIST BINARY
1668 AB QLE2
1669 EJECT
1670 *
1671 QLLIST TM LISTING ROUTINE
1672 BO ACTIVE+1,X'02'
1673 * QLLISTA,FORMATED LIST
1674 ST UNFORMATED LIST
1675 ST 14,SAR3
1676 DOIO WPKOUT,CARDIN
1677 BR 14,SAR3
1678 *
1679 *
1680 *
1681 *
1682 QLLMOV USE TABOVFL TO DD FOLD OVER
1683 MVC 0(0,REG5),0(REG4)
1684 LISTLN DS CLBO
1685 TABOVFL DS 200,H STR,LEN TARGET,LEN
1686 SAR3 DS 3F
1687 QLLISTA STM REG3,REG4,SAR3+4
1688 ST 14,SAR3
1689 ST 14,SAR3+4
1690 MVC LISTLN(11),CARDIN MOVE ACCESS NUMBER IN
1691 LA CNT,TABSS
1692 LA REG4,TABOVFL
1693 LA REG2,QTAB
1694 QLL1 A REG2,=F'16'
1695 A REG2,=F'16' CREATE OVERFLOW TABLE
1696 A REG4,=F'0'
1697 CLI QFS#2,X'CO' TEST IF ACTIVE
1698 BZ QLL2
1699 MVC 0(4,REG4),0(REG2) MVC LDC & LEN OF QUES
1700 L REG5,10(REG2)
1701 AR REG5,CNT
1702 MVC 4(4,REG4),0(REG5)
1703 QLL2 TM 5(REG2),X'01' TEST IF LAST QUES.
1704 BZ QLCI
1705 * MVI 0(REG4),X'FF' SET FLAG FOR END-HIGH ORDER
BYTE OF STR LDC AFTER LAST ONE

```

07/05/73

ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

```

1706 QLL7 MVC LISTLN(80),BLKS
1707 LA REG2,TABOVFL-8
1708 LA CNT,TABOVFL
1709 * REG2 NOW BASE FOR TABOVFL
1710 QLL3 A REG2,=F'8'
1711 TM 2(REG2),X'FF'
1712 BO QLL4 LAST QUES
1713 CLC 2(2,REG2),4(REG2)
1714 BNH QLL5
1715 LH REG3,6(REG2)
1716 LH REG4,2(REG2) TARGET SHORTER
1717 SRJ REG4,REG3
1718 MVC 10(8,REG2),0(CNT) COMPRESS TABOVFL
1719 STH REG4,2(CNT) STORE EXCESS
1720 A CNT,=F'8'
1721 B QLL6
1722 QLL5 LH REG3,2(REG2) INPUT SHORTER
1723 QLL6 BCTR REG3,0 MOVE STR
1724 * CORRECT LENGTH FOR EX OR MVC
1725 L REG5,4(REG2) LOAD TARGET LOC
1726 CL REG5,=F'80' TEST IF WITHIN PRINT LINE
1727 BH QLL6NG
1728 LA REG5,LISTLN(REG5)
1729 LA REG4,DIRREG2) LOAD INPUT LOC
1730 LA REG4,CARDLN(REG4)
1731 EX REG3,0(BMOV) MVC
1732 B QLL3
1733 QLL4 DOIO WORKOUT,LISTLN
1734 MVI 10(CNT),X'FF'
1735 CL CNT,=A(TABOVFL)
1736 BH QLL7
1737 QLLRTN4 LLM REG3,REG4,SAR3+4
1738 L 14,SAR3
1739 BR 14
1740 QLL6NG DOIO SYSPRINT,LISTERLN
1741 B QLLRTN4
1742 LISTERLN DC CLBO** LIST REQUESTED TAB SETTINGS TOO WIDE FOR PRIN
1743 NT LINE
1744 EJECT
1745 * BINARY TALLY ROUTINE
1746 * REQUIRES LIST-BTL AT A(WORKA) + C(Q03)+2
1747 * AND COUNTER LIST-CL OFFSET FROM BEGINNING OF BTL
1748 * STORED IN THE LAST H OF BTL
1749 * COUNTS-BIT1-99'S
1750 * BIT2-NR'S
1751 * BIT3-31-VALUES (29OF THEM)
1752 * BIT 32-NOT RECOGNIZED (WIERD)
1753 *LIST H(N-NUM OF VALUES),F(VALUE-BIT FLAGS),A(COUNTER STRING)
1754 *COUNTER LIST H(NR), (NH, H(WIERD,99)
1755 QLTALLY SAVE (14,12)
1756 L BR,=F'1' INCREMENT
1757 LA REG2,QTAB
1758 A REG2,=F'16'
1759 QLTT1 A REG2,=F'16'
1760 LH REG4,Q03

```

07/05/73

ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

```

1761      LA      REG4,WORKA(REG4)          REG4=A(LIST-BTL)
1762      LH      CNT,2(REG4)      CNT=VALUE BEING CHECKED
1763      L       REG5,4(REG4)     REG5=VALUE
1764      AH      REG4,8(REG4)     REG4=A(COUNTERS)
1765      *
1766      *      TEST NR,BIT FIRST-IFL-COUNTER RETURN
1767      L       TALLY,WEITHER ACTIVE GR NOT
1768      NR      REG3,=XL4'40000000
1769      BZ      REG3,REG5
1770      TLOP
1771      LH      REG3,0(REG4)
1772      AR      REG3,BR
1773      STH     REG3,0(REG4)
1774      B      TNO99
1775      LR      REG3,BR
1776      SLL     REG3,0(CNT)
1777      NR      REG3,REG5
1778      BZ      TNO
1779      LH      REG3,0(REG4)      BC IF BIT NOT SET
1780      AR      REG3,BR
1781      STH     REG3,0(REG4)      INCREMENT COUNTER
1782      A      REG4,=F'2
1783      *      CNT,TLOP
1784      LR      REG3,BR
1785      NR      REG3,REG5
1786      DZ      TNO99
1787      LH      REG3,0(REG4)
1788      AR      REG3,BR
1789      STH     REG3,0(REG4)      INCREMENT WIERO COUNT
1790      A      REG4,=F'2
1791      *      TEST 99 BIT
1792      L       REG3,=XL4'80000000
1793      NR      REG3,REG5
1794      BZ      TNO99
1795      LH      REG3,0(REG4)
1796      AR      REG3,BR
1797      STH     REG3,0(REG4)      INCREMENT 99 COUNT
1798      TM      QFS,1,X'01
1799      BZ      QLT1
1800      RETURN (1,12)      ENOUGH ALREADY
1801      *      EJECT
1802      *      BINARY LIST ROUTINE
1803      QLBLIST DC      CLBOX, 'A71XXXXX011 B NUM 32-1'S
1804      QLIST  DS      F
1805      QLBINARY SAVE (14,12)
1806      LA      REG2,QTAB1
1807      A      REG2,=F'16
1808      MVC     QLBLIST(10),CARDIN
1809      MVI     QLBLIST+10,C'B
1810      MVI     QLBLIST+18,C'O
1811      MVC     QLBLIST+19(31),QLBLIST+18
1812      QBIN2  A      REG2,=F'16
1813      LH      BR,003
1814      IC      REG4,=C'1
1815      *

```

352

07/05/73

***** ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

```

1816      LA      REG3,WORKA(BR)
1817      *      (ACTUALLY FLAGS)
1818      L        BR,4(REG3)
1819      L        CNT,=F'31'
1820      QBIN3   L        REG3,=F'1'
1821      SLA     REG5,0(CNT)    LOOP TESTS BIT0-30
1822      NR      REG3,BR
1823      BZ      QBIN1
1824      STC     REG4,QLBLIST+18(CNT)
1825      QBIN1  BCT     CNT,QBIN3
1826      L        REG5,=F'1'    TEST BIT 31
1827      NR      REG5,BR
1828      BZ      QBIN4
1829      STC     REG4,QLBLIST+18
1830      QBIN4  LH      REG4,QNUM    QUESTION NUMBER
1831      CVD    REG4,PACKED
1832      UNPK   QLBLIST+14(8),PACKED
1833      OI     QLBLIST+15,X'F0'
1834      LH     BR,2(REG2)    NUM OF POSS
1835      CVD    BR,PACKED
1836      UNPK   QLBLIST+51(2),PACKED
1837      OI     QLBLIST+52,X'F0'
1838      DOIO   MASPRT,QLBLIST
1839      TM     'ACTIVE+1,X'01'
1840      BZ      QBIN2
1841      RETURN (14,12)
1842      EJECT
1843      *
1844      *      *      *      *      *      *      *      *      *      *
1845      *      *      *      *      *      *      *      *      *      *
1846      *      *      *      *      *      *      *      *      *      *
1847      *      *      *      *      *      *      *      *      *      *
1848      *      *      *      *      *      *      *      *      *      *
1849      *      *      *      *      *      *      *      *      *      *
1850      *      *      *      *      *      *      *      *      *      *
1851      *      *      *      *      *      *      *      *      *      *
1852      *      *      *      *      *      *      *      *      *      *
1853      *      *      *      *      *      *      *      *      *      *
1854      *      *      *      *      *      *      *      *      *      *
1855      *      *      *      *      *      *      *      *      *      *
1856      *      *      *      *      *      *      *      *      *      *
1857      *      *      *      *      *      *      *      *      *      *
1858      *      *      *      *      *      *      *      *      *      *
1859      *      *      *      *      *      *      *      *      *      *
1860      *      *      *      *      *      *      *      *      *      *
1861      *      *      *      *      *      *      *      *      *      *
1862      *      *      *      *      *      *      *      *      *      *
1863      *      *      *      *      *      *      *      *      *      *
1864      *      *      *      *      *      *      *      *      *      *
1865      *      *      *      *      *      *      *      *      *      *
1866      *      *      *      *      *      *      *      *      *      *
1867      *      *      *      *      *      *      *      *      *      *
1868      *      *      *      *      *      *      *      *      *      *
1869      *      *      *      *      *      *      *      *      *      *
1870      *      *      *      *      *      *      *      *      *      *

```


07/05/73

ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

```

1871      B      QLCOM
1872      NI      ACTIVE+1,X'7F'
1873      TM      ACTIVE,X'03'
1874      BZ      QLFAND
1875      BM      QLFOR
1876      SRU     CNT,1(0)
1877      IFXOR2  NR      CNT,1
1878      BNZ     IFXOR1
1879      IFXOR4  SRU     CNT,1(0)
1880      LTR     CNT,CNT
1881      BZ      IFGO
1882      B      IFXOR2
1883      TM      ACTIVE+1,X'80'
1884      BNZ     IFNOGO
1885      NI      ACTIVE+1,X'80'
1886      B      IFXOR4
1887      *
1888      QLIFOR  D      1,=F'0'
1889      BNZ     IFGO
1890      B      IFNOGO
1891      *
1892      QLIFAND OR      1,CNT
1893      SLL     CNT,1(0)
1894      LTR     CNT,CNT
1895      BNZ     QLIFAND
1896      X      1,=F'-1'
1897      BZ      IFGO
1898      NI      ACTIVE+2,B'10101011'
1899      IFGO    RETURN  (14,12)
1900      LTR     LEAVE ALONE
1901      TITLE  CSECT  'BINARY TO INTEGER*2 FOR FORTRAN OUTPUT ROUTINE'
1902      EVQFBITS CSECT
1903      *
1904      * PASSED REG1 POINTS TO ADDRESSES
1905      * ADR 1 IS FULL WORD OF BIT FLAGS (PUT IN)
1906      * ADR 2 IS 32 HALF-WORD INTEGERS (REG2)
1907      BITS   SAVE   (14,12) (REG3)
1908      LR     BR1,15
1909      USING  EVQFBITS,BR1
1910      L      CNT,=F'31'
1911      L      REG2,0(1)
1912      L      REG3,4(1)
1913      * SHIFTS TO RIGHT- ARRAY(1) IS LEFT MOST BIT
1914      XC     0(64,REG3),0(REG3)
1915      B      BITSLP2
1916      A      REG3,=F'2'
1917      L      REG4,=F'1'
1918      SLL   REG4,0(CNT)
1919      N      REG4,0(REG2)
1920      BZ     BITSLP3
1921      MVC   0(2,REG3),=H'1'
1922      BCT   CNT,BITSLP
1923      TM    0(REG2),X'01'
1924      BZ     BITSLP4
1925      MVC   2(2,REG3),=H'1'
1926      RETURN (14,12)

```

TURN OFF CONDITIONAL BIT
CHECK OUT TOTAL CONDITIONAL

XOR TOTAL TEST
A ONE PRESENT

NO MORE TO TEST-NO REJECT-ACCEPT

TEST CONDITIONAL BIT
REJECT(1 ON)

FILL HIGH END OF 1 WITH 1'S

1 ALL 1'S
TURN 1,3,5 OFF

SET BIT TO BE TESTED
TEST INTO REGISTER

07/05/73

ASSEMBLER LISTING OF EVOS TERMINAL SYSTEM

PAGE 36

```

1926          LTORG
1927  EVOCST   BEGIN
1928          B   QTGEN
1929          *   CHAR STR TRANSLATOR (QTC)
1930  QTCCLC   GLC   010(REG4),1(REG5)
1931  QTGEN    LA   REG2,QTAD
1932  QTCLP    MVC   REG3,003      LOAD OFFSET IN QWRKA
1933          LA   REG3,WRKA(REG3)
1934          *   REG3 HAS A (TRANS. STR.)
1935          XC   4(REG3),4(REG3)  ZERO TALLY FLAGS
1936          LH   CNT,2(REG3)     LAOD COUNT OF POSS.
1937          LA   REG4,8(REG3)    LA OF 1ST POSS.
1938          LI   REG5,QLOG      L LOC OF QUES IN QWRKIN
1939          LA   REG5,CARDIN(REG5) LA QUES
1940          LA   BR,QTGEN
1941          *   CHK FOR NR.
1942          CLC   QLEN=F*0
1943          BE   QNR
1944          *   CHK FOR 99
1945          CLC   A:1(2(REG5),C+99
1946          BE   Q99
1947  QTCLP3   LH   BR,0(REG3)     L LEN OF POSS.
1948          EX   BR,QTCCLC
1949          BE   QT8
1950          AR   REG4,ARI        TRY NEXT ONE
1951          BCT  CNT,QTCLP3     BR BACK
1952          *   FES THRU LOOP *WIERD* ANSWER
1953          LA   BR,QTEND
1954          B   QWIERD
1955  QTCLP2   LH   BR,1
1956          SLA  BR,0(CNT)
1957          OR   BR,4(REG3)     SET FLAG
1958          ST   BR,4(REG3)     STORE FLAG
1959          TM   QFS+1,X'01'     SEE IF LAST QUESTION
1960          BZ   QTEND
1961          A   REG4,F*16
1962          B   QTCLP          DO NEXT QUESTION
1963  QTEND
1964  QWMOV    BR   14
1965          MVC   WOUT,18(0),1(REG5)
1966          *   RESULT OF QTC
1967          *   NTH MATCH SETS BIT CNT-N+2 TO 1
1968          *   NO MATCH SETS BIT 1 TO 1
1969          *   EX MATCH BRD STR-SET7
1970          *   COUNT8 = 8-3+2=7
1971  Q99      TM   ACTIVE,X'80'
1972          BZ   QW1
1973          DI   4(REG3),X'80'   SET TALLY FLAG
1974          MVC   WOUT(80),BLKS
1975          MVI  WOUT+10,C'9'    Z=9FOR99
1976  QW2     LH   CNT,QLEN      L LEN OF QUESTION
1977          EX   CNT,QWMOV     MOVE QUES TO MSG STR
1978  QW3     MVC   WOUT(11),CARDIN
1979          L   CNT,ONUM
1980          CVD  CNT,PACKED
          UNPK  WOUT+14(3),PACKED

```

355

07/05/73

ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

PAGE 37

```

1981      OI      WOUT+16,X*F0*
1982      ST      14,0(13)
1983      OOI0    MASPRT,WOUT
1984      *
1985      L        14,0(13)      SET TALLY FLAG
1986      OW1     BR      BR
1987      WOUT    DC      CLB0, 'A71XXXXX01Z' NUM (QUES)-
1988      *
1989      QNTRD   TM      ACTIVE,X*40*
1990      BZ      OW1
1991      MVC     WOUT(B0),BLKS
1992      MVI     WOUT+10,C'W'      Z=W FOR WIERD
1993      OI      7(REG3),X*01*    SET TALLY FLAG
1994      BZ      QW2
1995      *
1996      *
1997      *
1998      *
1999      *
2000      QNR     TM      ACTIVE,X*10*
2001      BZ      OW1
2002      MVC     WOUT(B0),BLKS
2003      MVI     WOUT+10,C'N'
2004      OI      7(REG3),X*40*    SET TALLY FLAG
2005      BZ      QWB
2006      EVQBGN LTORG
2007      LA     COMBASE,BGNCOMS
2008      L      REG2,COMROUT
2009      BR      REG2
2010      BGNCOMS DC      X'0802080405'
2011      BLETS  DC      0,ELFHP
2012      BNUML  DC      F'5'
2013      BRETNS DC      X'0802080405'
2014      BADRS  DC      A(E1,W2D,L3D,F1,P2,1,P3)
2015      B0FSTS DC      X'00G00C000C'
2016      SAVEF  DS      1,BF
2017      PAPMLST DS      2,F
2018      DC      X'00',AL3(BZRD)
2019      PICA   DS      F
2020      BZRD   DC      F'0'
2021      E1     SAVER  SAVE=SAVEF
2022      CALL   EVQERR
2023      ST     1,PICA
2024      L     13,SAVEF+4
2025      RETURN (14,12)
2026      INTVL  DC      R10,H'0'
2027      W2D    L      REG2,PARM
2028      PACK   INTVL(3),1(4,REG2)
2029      MVO    INTVL+1(3),INTVL(3)
2030      MVI    INTVL+3,X'00'
2031      STIMER WAIT,DINTVL=INTVL
2032      BR     14
2033      BMOV   MVC     BNAM(0),V1(REG3)
2034      BNAM   DS      CLB0
2035      BPARAM DS      H,CL70

```

356

07/05/73

***** ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

PAGE 38

```

2036 L3D MVC BNAM,BLKS
2037 SAVER SAVE=SAVEF
2038 L REG2,PARML
2039 A REG2,=F'-1
2040 L REG3,PARM
2041 EX REG2,BMOV
2042 XC REG2,REG2
2043 LR REG3,CHAR
2044 A REG3,=F'1
2045 L3A CLI DTCHAR,C
2046 RE L38
2047 A CHAR,=F'1
2048 A REG2,=F'1
2049 B L38
2050 L38 C REG2,=F'0
2051 BH L3C
2052 LA 1,PICA-4
2053 LINK EPLOC=BNAM
2054 LA 13,SAVEF+4
2055 RETURN (14,12)
2056 L3C A REG2,=F'-1
2057 STH REG2,UPARM
2058 A REG2,=F'-1
2059 EX REG2,BMOV
2060 LINK EPLOC=BNAM,PARAM=(BPARAM),VL=1
2061 LA 13,SAVEF+4
2062 RETURN (14,12)
2063 F1 L 2,PICA
2064 SPIE MF=(E,(2))
2065 BR 14
2066 P2 STM 0,13,SAVEF+4
2067 LA REG2,SAVEF+4
2068 ST REG2,PARMLST
2069 LA REG2,PARMLST-4
2070 ST REG2,PARMLST+4
2071 LA 1,PARMLST
2072 CALL POUND,ID=4
2073 LM 0,13,SAVEF+4
2074 BR 14
2075 P3 SAVER SAVE=SAVEF
2076 L REG2,PARM+4
2077 MVC EPC,1(REG2)
2078 BAL 14,EPHEX
2079 ST 2,PARMLST
2080 L REG2,PARM
2081 MVC EPC,1(REG2)
2082 BAL 14,EPHEX
2083 ST 2,PARMLST+4
2084 LA 1,PARMLST
2085 CALL POUND,ID=5
2086 L 13,SAVEF+4
2087 RETURN (14,12)
2088 BGD BR 14
2089 EPHEX MVC EPC,1(REG2)
2090 TR EPC,TABLE

```

INDICATE NJ PARM LIST

357

07/05/73

***** ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

PAGE 39

```

2091      PACK      HEX-1, EPC
2092      MVO       HEX+1(4), HEX-1(4)
2093      MVI       HEX, X'00'
2094      L         REG2, HEX
2095      BR        14
2096      DS        OF, CL2
2097      EPC      DS        CL6
2098      HEX     DC        XL4, X'00'
2099      TABLE  DC        193C
2100      DC        X'0A0B0C0D0E0F'
2101      DC        1C
2102      DC        C'0123456789'
2103      DC        CL6
2104      LTORG
2105      EJECT
2106      END
2107      EVOERR   CSECT
2108      REG2    EQU      12
2109      BRI      EQU      11
2110      SAVE    (14, 12)
2111      USING   EVOERR, 11
2112      LR      11, 15
2113      *      ERROR ROUTINE
2114      *      COMMANDS START IN COL 1
2115      *      D 000000 000000 DUMP A(FROM) A(TO)
2116      *      PSW 000000 PSW A(RESTART)
2117      SPTF    ERRR, ((1, 15))
2118      ST      1, PICE
2119      LM      14, 12, 12(13)
2120      L       1, PICE
2121      BR      14
2122      BR      14
2123      ERRR    EQU      *
2124      USING   ERRR, 15
2125      STM     14, 13, SAVEP
2126      MVC     OPSW, 4(1)
2127      MVC     RONE, 12(20(1))
2128      MVC     REND, 12(1)
2129      STM     3, 13, RMID+4
2130      DRDP    15
2131      LR      BR1, 15
2132      USING   ERRR, BR1
2133      LA      13, SAVE
2134      MVC     EPARM, =A(OPSW)
2135      MVC     EPARM+4, =A(EPARM)
2136      OPEN   (ERRIN)
2137      LA      1, EPARM
2138      CALL   PDUMP, ID=2
2139      IN     READ  ERRDECB, SF, ERRIN, CARD, 80
2140      CHECK  ERRDECB
2141      CLI    CARD, C'D'      D=DUMP
2142      MVC     EPC, CARD+2
2143      BNE   CHA
2144      BAL    14, EPHEX
2145      ST     2, EPARM

```

07/05/73

ASSEMBLER LISTING OF EVQS TERMINAL SYSTEM

```

2146      MVC      EPC,CARD+9
2147      BAL      #4,EPHEX
2148      ST       #2,EPARM+4
2149      BR       DMP,1
2150      *
2151      EPHEX    TR      EPC,TABLE
2152      MPACK   TR      HEX-1,EPC
2153      MVU    TR      HEX+1(4),HEX-1(4)
2154      MVL    TR      HEX,X'00'
2155      L      RFG2,HEX
2156      BR     #4
2157      *
2158      CHA     CLI     CARD,C+P
2159      BNE     OVER
2160      MVC     EPC,CARD+4
2161      LA     #4,OR
2162      B      EPHEX
2163      OVER   CLI     CARD,C+P
2164      BE     IN
2165      OV2    L      #2,PICE
2166      SPIE   MF=(E,(2))
2167      DROP   BR1
2168      USING  ERRR,15
2169      LR     #15,BR1
2170      LM     #4,12,SAVEP
2171      DR     #4
2172      OR     LR     #15,BR1      RESTORE REG 15
2173      LM     #4,12,SAVEP      RESTORE REGS
2174      MVC     #9(3,1),HEX+1
2175      BR     #4
2176      CARD   DS     CL80
2177      DS     OF,CL2
2178      EPC    DC     CL6
2179      HEX    DC     XL4'00',X'00'
2180      TABLE DS     193C
2181      DC     X'0A0B0C0D0E0F'
2182      DC     41C
2183      DC     C'0123456789'
2184      DC     CL6
2185      SAVEP  DS     18F
2186      ADMP   DS     F
2187      PICE   DS     F
2188      ERRIN  DCB    DDNAME=ERRIN,BLKSIZE=80,BUFNO=1,DSORG=PS,
2189      EODAD=OV2,LRECL=80,MACRF=(R),RECFM=F
2190      EZD    DC     F'0'
2191      SAVE   DS     18F
2192      QPSW  DS     FL8
2193      RONE  DS     FL8
2194      RMID  DS     12FL4
2195      REND  DS     FL8
2196      EPARM DC     A(OPSW)
2197      DC     A(EPARM-4)
2198      DC     X'80',AL3(EZD)
2199      LTORG

```

07/05/73

THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55

***** THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

*** FORTRAN ROUTINE FOR TALLY OUTPUT

```

SUBROUTINE EVQSFORPT(QTAB,WORKA,IFLIST,IFNUM)
  IMPLICIT INTEGER*2 (A-Z)
  INTEGER*2 QTAB(8,20),WORKA(400),IFLIST(4,15),LOGIC(32)
  INTEGER IFNUM
  DO 100 I=1,20
    CALL EVQFBITS(QTAB(4,I),LOGIC)
    IF (LOGIC(16).EQ.1) GO TO 200
    IF (LOGIC(15).NE. 1) GO TO 150
    WRITE (6,900)
    900 FORMAT('      NEXT LINE')
    QNUM=QTAB(5,I)
    Q01=QTAB(6,I)
  END
  Q03=QTAB(8,I)
  CNTN=WORKA(Q03/2+2)
  CNT1=(WORKA(Q03/2+5) + Q03 +3) / 2
  CNTL = CNT1 - CNTN
  150 WRITE(6,910) (QNUM,Q01,(WORKA(J),J=CNT1,CNTL))
  910 FORMAT(5X,2I10,10I10/25X,10I10)
  CONTINUE
  CONTINUE
  DO 300 I=1,IFNUM
    300 WRITE(6,920) I,IFLIST(2,I)
  920 FORMAT(10X,2I10)
  STOP
  END

```

** PREPNH PROGRAM

```

PREPNH START
* PARAMETERS ARE ENTERED PARM='00010,00020'
* OR IF 'LOADER' IS USED PARM='00010,00020'
SAVER SAVE (14,12),*
BALR 2,0

```

A.2 Utility Programs

07/05/73

THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

PAGE 2

```

56 USING 2,2
57 3,0,1)
58 MVC INPUT(11),2(3)
59 OPEN (OUTPUT,OUTPUT)
60 PACK START,INPUT
61 PACK STOP,IN2)
62 MVC INSEQ,START
63 PUNCH UNPK SEQ,INSEQ
64 DI SEQ+4,X'F0'
65 PUT OUTPUT,CARD
66 AP INSEQ,STEP
67 CLC INSEQ,STOP
68 BNH PUNCH
69 CLOSE (OUTPUT)
70 RETURN (14,121,RC=999)
71 * DCB -- INFO FILLED FROM DD CARD
72 OUTPUT DC A(0,0,0,0,0,1,16384,1,1)
73 DC F'0' RECFM,BLKSIZE,LRECL
74 DC CL8'FT07F001' ON DD CARD
75 DC X'02005050' OR ON VOLUME VTOC
76 DC A(1,1,0,0,1,1,1,0,1,0,1)
77 DS OD
78 INSEQ DS PL4
79 STEP DC PL4,10
80 START DC PL4,10
81 STOP DC PL4,90
82 INPUT DC ZL5,10
83 DC CL1
84 IN2 DC ZL5,80
85 CARD DS OCL80
86 DC C
87 SEQ DC ZL5,0
88 DC 73C
89 END PREPNH

```

** OBCORR

```

101 MACRO
102 &NAM QDCB &DDNAME=, &LRECL=80, &BLKSIZE=80, &RECFM=F, &EODAD=1
103 * ONLY RECFM=F OR FB ALLOWED - ALL ELSE LEAVES DCB FIELDS ZEROED
104 &NAM DC A(0,0,0,0,0,1,16384,1,&EODAD)
105 AIF( ('&RECFM' EQ 'F') ,F
106 DC F'0' RECFM,BLKSIZE,LRECL
107 DC CL8,&DDNAME, ON DD CARD
108 DC X'02005050' OR ON VOLUME VTOC
109 DC A(1,1,0,0,1,1,1,0,1,0,1)
110 MEXIT

```


07/05/73

THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

```

111 .F DC BL1,10000000,XL3'00' RECFM=F
112 .REST DC CLB,DDNAME
113 DC X'02005050'
114 DC A(1,1,&BLKSIZE,0,1,1,1,&LRECL,1,0,1)
115 MEND
116 OBCORR CSECT
117 REG2 EQU 2
118 REG3 EQU 3
119 CHAR EQU 10
120 BASE EQU 12
121 SAVE (14,12)
122 USING OBCORR,BASE
123 LR BASE,15
124 L REG2,0(1)
125 OPEN (SYSIN,INPUT,SYSPRINT,OUTPUT,EVQSOUT,OUTPUT)
126 RD MVC XEOL,BLKS
127 A REG,ONE
128 MVC XEOL,BLKS
129 GET SYSIN,CARD
130 LA CHAR,CARD+10 LA OF DATA FIELD
131 LOOP A CHAR,ONE
132 CLI 0(CHAR),X'FE'
133 BC 8,EOL
134 CLI 0(CHAR),C'@' LOOK FOR EOH
135 BC 8,EOL BC ON=
136 CLI 0(CHAR),C'@' LOOK FOR EOL (0.8.2)
137 BC 6,LOOP BC ON NOT =
138 BC 8,PRNT BC ON=
139 EOL MVC XEOL,AEOL
140 PRNT PUT SYSPRINT,LINE
141 PUT EVQSOUT,CARD
142 BC 15,RD
143 EOH CLI 1(CHAR),C'@'
144 BC 8,EOH1
145 MVI 0(CHAR),C'#'
146 BC 15,LUOR
147 EOH1 MVC XEOL,AEOL
148 BC 15,PRNT
149 ENDFILE CVD REG3,TOTNUM
150 UNPK OUTNUM,TOTNUM
151 DI OUTNUM#3,X'F0'
152 PUT SYSPRINT,FINELINE
153 CLOSE (SYSIN,,SYSPRINT,,EVQSOUT)
154 RETURN (14,12),RC=999
155 EVQSOUT QDCB DDNAME=EVQSOUT,RECFM=
156 SYSPRINT QDCB DDNAME=SYSPRINT,BLKSIZE=132,LRECL=132
157 SYSIN QDCB DDNAME=SYSIN,EODAD=ENDFILE,RECFM=
158 TOTNUM DS
159 ONE DC F'1'
160 LK DC X'00'
161 ZERO DC F'0'
162 FINELINE DS OCLBO
163 DC
164 OUTNUM DS ZL4
165 DC SC' C CARDS PROCESSED,CL70'

```

362

07/05/73

THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

PAGE 4

```

166  LTNE  DS  OCL13B
167  DC  LIC
168  XEOL  DC  CL6
169  DC  100
170  XEDH  DC  CL8
171  DC  100
172  CARD  DC  CL80
173  DC  BXFF
174  DC  CL15
175  AEOL  DC  C AND EDU
176  AEOH  DC  C**EDH**
177  BLKS  DC  CL6
178  END  OBCOOR
179
180
181
182
183
184
185
186  ** VERIFY PROGRAM VRFY
187
188
189
190
191  VRFY:  PROC  OPTIONS(MAIN);
192  DCL  CARD CHAR(80) VAR,
193  EOH CHAR(2) INITIAL('00'),
194  TABH(4) INITIAL(1,20,40,80);
195  DCL  SPLIT(20,2) BIN,
196  EQO CHAR(1) INITIAL(' '),
197  EOL CHAR(1) INITIAL(' ') /* 0.8.2*/ ,
198  (TABS,TABS2)(10,20) BIN INITIAL( (200) 8) ,
199  CARDS(15) CHAR(80),LINE CHAR(120),
200  (IS,TAB) FIXED,EOLERR BIT(1),SEQS(8),
201  S1 BIT(1), (CNT,CNT1,CNT2) BIN ,DTE CHAR(6),
202  TAB(10) BIN INITIAL( (10) 10),
203  DATALINE CHAR(240) VAR,
204  DELIMITER CHAR(2) VAR INITIAL(' ');
205  DCL NOTE CHAR(2) INITIAL(' ');
206  DCL CONTROL FILE;
207  /* TABS(TYPE,COLM) IS COLUM WIDTH */
208  /* TABS2(TYPE,COLM) IS COLUM STARTING ELEMENT */
209  OPEN FILE(SYSPRINT) OUTPUT STREAM PRINT LINESIZE(132);
210  ON ENDFILE(SYSPRINT) BEGIN;
211  PUT FILE(SYSPRINT) DATA(NUM CARDS) SKIP(10);
212  STOP;
213  END;
214  ON ERROR SNAP BEGIN;
215  ON ERROR SNAP CALL IHEDUMP;
216  PUT FILE(SYSPRINT) SKIP(3);
217  PUT LIST('ONCODE= ',ONCODE);
218  PUT DATA(CNT1,CNT2,CNT);
219  PUT DATA(NUM CARDS);
220  PUT DATA(T1,T2,TYPE,J);
221  PUT DATA(LINE,CARD);

```

363

07/05/73

THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

PAGE 5

```

221      PUT DATA(TAB2,TAB1,TABA)
222      PUT DATA(SPLIT)
223      PUT DATA(DATALINE) SKIP
224      STOP
225      END
226
227      EOLERR=0
228      PAG=0
229      NUM CARDS=0
230      IS=0
231      I=0
232      TAB2=0
233      TYPE=0
234      TAB1=0
235      TABA=0
236      LINE=
237      DTE=DATE
238      GET FILE(CONTROL) DATA(TAB,TABS,PAG)
239      PUT FILE(SYSPRINT) DATA(TAB)
240      PUT FILE(SYSPRINT) EDIT('TABS -BY TYPE VERTICALLY ',TABS)
241      (SKIP,A,(120) (SKIP,(10) F(7)))
242      PUT FILE(SYSPRINT) DATA(PAG) SKIP
243
244      -LP1:
245      DO IK=1 TO 10
246      S1='1'B
247      TOT=1
248      DO IL=1 TO 20
249      TOTA=TABS(IK,IL)
250      TABSS(IK,IL)=TOT
251      TOT=TOT+2*TABS(IK,IL)
252      IF(TOT>120) THEN
253      IF(S1) THEN DO
254      S1='0'B
255      PUT FILE(SYSPRINT) EDIT('COLM',IL,'TYPE',IK,
256      ENDS IN PRNT COLM ',TOT) (A,F(4),A,F(4),A,F(6)) SKIP
257      END
258      ENDDL1
259      OREG: GET FILE(SYSIN) EDIT(CARD) (A(80))
260      OSTART: IF(INDEX(CARD,FOH)=0) THEN
261      DO
262      PUT FILE(SYSPRINT) SKIP
263      EDIT('FIRST CARD ISNT A HEADER CARD **',CARD)
264      (A,A(80))
265      NUM CARDS = NUM CARDS + 1
266      GO TO BEG
267      END
268      SEQ=SUBSTR(CARD,1,11)
269      CARD=SUBSTR(CARD,12)
270      S1='1'B
271      T1=0
272      LINE=
273      DO J=1 BY 1 WHILE (S1)
274      T2=INDEX(CARD,SEQ)
275      IF(T2=0) THEN DO
276      T2=INDEX(CARD,EOH)

```

07/05/73

THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

PAGE 6

```

276         S1='0'B ;
277     END ;
278     IF (J#B) THEN TYPE= SUBSTR(CARD,T1+1,T2-T1-1) ;
279     SUBSTR(LINE,TABH(J)) = SUBSTR(CARD,T1+1,T2-T1-1) ;
280     T1=T2 ;
281     SUBSTR(CARD,T1,1) = ' ' ;
282     END ;
283     PAG=PAG+1 ;
284     PUT FILE(SYSPRINT) EDIT('TIME ',TIME)(PAGE,COLUMN(110),A,A(9)) ;
285     PUT FILE(SYSPRINT) EDIT('DATE ',DTE)(SKIP,COLUMN(110),A,A(6)) ;
286     PUT FILE(SYSPRINT) EDIT('PAGE ',PAG)(SKIP,COLUMN(110),A,A(4)) ;
287     PUT FILE(SYSPRINT) EDIT('LINE,SEQ)
288     (SKIP,A(120),X(2),F(8)) ;
289     - NLINE=0 ;
290     SEQ=0 ;
291     DATALINE=' ' ;
292     -LOOP: DO ICARD=1 BY 1 ;
293     GET FILE(SYSIN) EDIT(CARD)(A(80)) ;
294     IF (INDEX(CARD,EOR)>0 && NLINE>7) THEN DO ;
295     PUT FILE(SYSPRINT) EDIT((CARDS(J) DO J=1 TO ICARD-1)
296     (SKIP(5), (15) (COLUMN(40),A(80),SKIP)) ;
297     NUM CARDS=NUM CARDS + ICARD ;
298     NLINE=0 ;
299     GO TO START ;
300     END ;
301     0. CARDS(ICARD)=CARD ;
302     PRE=SEQ ;
303     SEQ=SUBSTR(CARD,1,8) ;
304     IF (SUBSTR(CARD,12,1)=' ') THEN ZAP=13 ;
305     ELSE ZAP=12 ;
306     - /* TAB2= PRNT COLM LENGH ;
307     TAB3 = DATA COLM LENGTH */ ;
308     /* SPLIT(I,1)= STARTING ELEMENT OF QUES. I */ ;
309     /* SPLIT(I,2)= LENGTH OF QUES. I */ ;
310     ENDCARD = INDEX(CARD,EOL) ;
311     - IF (ENDCARD=0) THEN
312     DO ;
313     DATALINE=DATALINE || SUBSTR(CARD,ZAP) ;
314     GO TO END_LOOP ;
315     END ;
316     NLINE=NLINE+1 ;
317     PUT FILE(SYSPRINT) SKIP ;
318     -
319     DATALINE=DATALINE || SUBSTR(CARD,ZAP,ENDCARD-ZAP+1) ;
320     LDL=LENGTH(DATALINE) ;
321     CNT2=INDEX(DATALINE,EOL) ;
322     CNT1=1 ;
323     S1='1'B ;
324     DO CNT=1 BY 1 ;
325     SPLIT(CNT,1)=CNT1 ;
326     SPLIT(CNT,2)=CNT2-1 ;
327     IF (CNT2=0) THEN GO TO CNT2 ;
328     CNT1=CNT1+CNT2 ;
329     CNT2=INDEX(SUBSTR(DATALINE,CNT1),EOL) ;
330     END ;

```

365

07/05/73

THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

PAGE 7

```

331 0 CONT2: TABA=CNT /
332 SPLIT(CNT,2)=LDL-CNT1 /
333 IF(TABA>TAB(TYPE) & TABA>20) THEN EOLERR='1'B /
334 NEWLINE /
335 LINE=' ' /
336 LENTOT=0 /
337 DO I=1 TO TABA /
338 TAB2=TABS(TYPE,I) /
339 TAB3=SPLIT(I,2) /
340 IF(TAB3=0) THEN GO TO ENDLINE /
341 IF(TAB2/TAB3) THEN DO /
342 SPLIT(1,2)=TAB3-TAB2 /
343 LENTOT=LENTOT+LEN /
344 LEN=TAB2 /
345 END /
346 0 ELSE DO /
347 SPLIT(I,2)=0 /
348 LEN=TAB3 /
349 END /
350 0 SUBSTR(LINE,TABS(TYPE,I),LEN)=
351 SUBSTR(DATALINE,SPLIT(I,1),LEN) /
352 IF(TAB2/TAB3) THEN
353 SPLIT(I,1)=SPLIT(I,1)+TAB2 /
354 -ENDLINE: END /
355 SEK=SEQ /
356 NOTE=' ' /
357 IF(SEQ)PRE) THEN NOTE='1' /
358 ELSE NOTE=' ' /
359 PUT FILE(SYSPRINT) EDIT(NOTE,SEK,LINE)
360 (SKIP,A(2),F(8),X(2),A(120)) /
361 0 IF(LENTOT>0) THEN GO TO NEWLINE /
362 IF(EOLERR) THEN DO /
363 Z=INDEX(CARD,EOL) /
364 PUT FILE(SYSPRINT) EDIT('**ERROR** EITHER MISSING EO
365 L OR TAB SETTINGS WRONG','*',CARD,'*',SEQ) /
366 (SKIP,A,SKIP,COLUMN(28),A(2),A,A(2),F(8)) /
367 IF(Z>0) THEN PUT FILE(SYSPRINT)
368 EDIT ('EOL IN LINE ABOVE','*') /
369 (SKIP,A,COLUMN(29+Z),A(1)) /
370 EOLERR='0'B /
371 END /
372 DATALINE=' ' /
373 - END-LOOP: END LOOP /
374 - END VREY /
375
376
377
378
379
380 ** RENUM
381
382
383
384 RENUM PROC OPTIONS(MAIN) /
385 DCL (OLD,SEQ) FIXED DEC (7),

```

366

07/05/73

THE SMALLER PROGRAMS , INCLUDING TALLY OUT ROJT

PAGE 8

```

386 DATA CHAR (69) , CARD CHAR (80) VAR ,
387 EOH CHAR (2) INITIAL ('@'),
388 EOP CHAR (1) INITIAL ('-'),
389 FORM CHAR (2) , PAGE FIXED DEC (2),
390 LAKES (4) CHAR (8) VAR INITIAL ('SCHROON', 'SARATOGA',
391 'ONEIDA', 'L.G. '),
392 DUMM CHAR (10) VAR, SW1 BIT (1) INITIAL ('1'B),
393 EDL CHAR (1) INITIAL ('1'), CARD2 CHAR (80) VAR,
394 (SW2, SW3) BIT (1),
395 (OLD_RESP, RESP) FIXED DEC (5), CARDNUM FIXED DEC (1),
396 DCL (SW4, SW5) BIT (4), TISC CHAR (1), FORM2 CHAR (3),
397 EOP CHAR (1) INITIAL ('-'), CARDS CHAR (690) VAR,
398 /* SW1 IS OFF (0) IF AT LEAST 1 HEADER CARD HAS BEEN FOUND
399 SW4 IS ON IF FORM TYPE IS MULTI-PART ON HEADER CARD */
400
401 SW2, SW3 = '1'B
402 0 GN ENDFILE (SYSIN) SNAP BEGIN
403 PUT DATA
404 STOP
405 END
406 0 ON CONVERSION SNAP BEGIN
407 ONCHAR = '0'
408 PUT DATA (SEQ, DATA)
409 END
410 - OLD_SEQ = 0
411 0 START: GET FILE (SYSIN) EDIT (SEQ, DATA) (X(2), F(7), X(2), A(69))
412 CNT = 0
413 0 IF (SEQ = 0) THEN DO
414 PUT FILE (SYSPRINT) EDIT ('*NOTE* BLANK CARD IN INPUT FILE')
415 (A) SKIP
416 GO TO START
417 END
418 0 IF (SEQ < OLD_SEQ) THEN LP1: DO
419 PUT FILE (SYSPRINT) EDIT ('**ERROR IN SEQUENCE NUMBER')(A) SKIP
420 0 BR1: PUT FILE (SYSPRINT) EDIT (SEQ, DATA) (X(2), F(7), X(2), A(69)) SKIP
421 DO I = 1 TO 20
422 GET FILE (SYSIN) EDIT (CARD) (A(80))
423 PUT FILE (SYSPRINT) EDIT (CARD) (A(80)) SKIP
424 END
425 STOP
426 END LP1
427 - IF (INDEX (DATA, EOH) > 0) THEN LP2: DO
428 DELIM = INDEX (DATA, EOP)
429 0 IF (PAGE = 1) THEN P1: DO
430 IND_EOP = INDEX (SUBSTR (DATA, 1, DELIM), EOP)
431 SW4 = '1'B
432 IF (IND_EOP > 0) THEN
433 FORM2 = SUBSTR (DATA, IND_EOP + 1, 2)
434 ELSE SW4 = '0'B
435 END P1
436 0 ELSE SW4 = '0'B
437 FORM4 = SUBSTR (DATA, 1, 2)
438 DELIM2 = INDEX (SUBSTR (DATA, DELIM + 1, EOP))
439 DUMM = SUBSTR (DATA, DELIM + 1, DELIM2 - 1)
440 DO I = 1 TO 4

```

07/05/73

THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

PAGE 9

```

441 IF(DUHM=LAKES(1)) THEN LAKE =1
442 END
443 DELIM2=DELIM2-DELIM1
444 DELIM=INDEX(SUBSTR(DATA,DELIM2),EQQ)
445 PAGE=SUBSTR(DATA,DELIM2,DELIM-1)
446 SW1='0'
447 PUT FILE(SYSPRINT) EDIT (FORM,LAKE,PAGE)
448 (COLUMN180),A(2),X(10),F(1),
449 X(10),F(2))
450 CARDS=1
451 GO TO START
452 END LP2
453 - IF(SW1) THEN DO
454 PUT FILE(SYSPRINT) EDIT (**ERROR** -- MISSING INITIAL HEADER
455 CARD) (SKIP(3),A)
456 PUT FILE(SYSPRINT) EDIT (SEQ,DATA) (X(2),F(7),X(2),A(69)) SKIP
457 GO TO START
458 END
459 - DATA CARDS
460 END=INDEX(DATA,EOL)
461 IF(END<69) THEN CARDS=CARDS+1 SUBSTR(DATA,1,END)
462 ELSE CARDS=CARDS+1 DATA
463 IF(END=0) THEN GO TO START
464 DELIM=INDEX(CARDS,EQQ)+1
465 0 IF(DELIM=2) THEN DO
466 CARDS=SUBSTR(CARDS,2)
467 DELIM=INDEX(CARDS,EQQ)+1
468 END
469 EOPDLM=INDEX(SUBSTR(CARDS,1,DELIM),EOP)
470 IF(EOPDLM>0) THEN DELIM=EOPDLM+1
471 RESP=SUBSTR(CARDS,1,DELIM-2)
472 0 IF(RESP=0) THEN DO
473 PUT FILE(SYSPRINT) EDIT (**CARD WITH BAD RESP. NO. - SKIPED*,
474 CARDS) (SKIP,A,(10) (SKIP,A(80)))
475 GO TO START
476 END
477 0 IF(SW4) THEN DO
478 DATA=FORM2+1 SUBSTR(CARDS,DELIM,66)
479 CARDS=SUBSTR(CARDS,65+DELIM)
480 END
481 0 ELSE DO
482 DATA=1 SUBSTR(CARDS,DELIM,68)
483 CARDS=SUBSTR(CARDS,DELIM+67)
484 END
485 CARDNUM=1
486 DO WHILE(LENGTH(CARDS)>0)
487 0 OTHERS
488 PUT FILE(SYSPUNCH) EDIT
489 (FORM,LAKE,RESP,PAGE,CARDNUM,DATA)
490 (A(2),F(1),P'99999',P'99',F(1),A(69))
491 CARDNUM= CARDNUM+1
492 DATA=SUBSTR(CARDS,1,69)
493 CARDS= SUBSTR(CARDS,70)
494 END
495 GO TO START

```

07/05/73

THE SMALLER PROGRAMS INCLUDING TALLY OUT ROUT

PAGE 10

496

END RENUM

07/05/73

SAMPLE LISTING OF EVQS DATA KEYPUNCH ERRORS *****

PAGE 1

1 *****
 2 75
 3
 4 75
 5
 6
 7
 8
 9 292 AK-SCHROON-3-YASH @@
 10 292 A5-SCHROON-3-YASH@@
 11
 12
 13
 14
 15
 16 364 100-3
 17 365
 18
 19 365
 20
 21
 22
 23
 24 381 5045-7
 25 382 5045-M#31#4#GOLF-F#30#4#6-F#7.5#4=6-F#3.5#4#
 26
 27 381
 28 382 5045-M#31#4#GOLF-F#30#4#6-F#7.5#4#6-F#3.5#4#
 29
 30
 31
 32
 33 510 LKESSE#
 34
 35 510 LK ESSE#
 36
 37
 38
 39
 40 529 563-1-1#2#99-2-2--150-1-17 65 08
 41 530 566-1-1#4-2-2--200-1-17 75 08#12 6 08#
 42
 43 529 563-1-1#12#99-2-2--150-1-17 65 08 #
 44 530 566-1-1#4-2-2--200-1-17 75 08#12 10 08 #
 45
 46
 47
 48
 49 543 571025-1-1956-2000--1-2-1-2-9#
 50
 51 543 571-25-1-1956-2000--1-2-1-2-9#
 52
 53
 54
 55

A.3 Sample Error Listings

 07/05/73 SAMPLE LISTING OF EVOS DATA KEYPUNCH ERRORS: *****

56	I/OJG6U0/LM-9#770-W SIDE SCHROON LK-1C#2E-9#10#70-9#16#70-S SCHROON
57	
58	2034#G1#3067-9#770-W SIDE SCHROON LK-1C#2E-9#10#70-9#16#70-S SCHROON
59	
60	
61	
62	
63	I/JJ
64	2043#G1#3076#NR-9#770-W SIDE SCHROON LK-1C#2E-9#8#70-9#11#70-S SC
65	HRDESSE
66	
67	
68	2043#G1#3076#NR-9#770-W SIDE SCHROON LK-1C#2E-9#8#70-9#11#70-S SCHRO
69	ON ESSE
70	
71	
72	
73	
74	2045#G1#3078-9#770-W SIDE SCHROON LK-1C#2E-9#7#70-10#6#70-W SHORE SC
75	
76	2045#G1#3078-9#770-W SIDE SCHROON LK-1C#2E-9#7#70-10#6#70-W SHORE SC
77	
78	
79	
80	
81	2061#G1#3094-9#770-W SIDE SCHROON LK-1C#2E-9#7#70-9#9#70-GROVE PT RD
82	
83	2061#G1#3094-9#770-W SIDE SCHROON LK-1C#2E-9#7#70-9#9#70-GROVE PT RD
84	
85	
86	
87	
88	2049-13-1-53-M-M-1-VOLLEY STREAM NY NASS-5-2
89	
90	2049-13-1-53-M-M-1-VALLEY STREAM NY NASS-5-2
91	
92	
93	
94	
95	2038-13-1-60-F-S-1-BROOKLYN NY KING-6-1
96	
97	2038-13-1-60-F-S-1-BROOKLYN NY KING-6-1
98	
99	
100	
101	
102	2045-1-1-1#99-2--125-1--60
103	
104	2045-1-1-1#99-2--125-1-60
105	2062-75-1-53-12000--3-3-1-2-9#12#20#99
106	
107	2062-75-1-1953-12000--3-3-1-2-9#12#20#99
108	
109	
110	

07/05/73

SAMPLE LISTING OF EVQS DATA KEYPUNCH ERRORS *****

PAGE 3

111	779	C4-SCHROON-. *MOM/-WELLS HOUSE HOTEL-1B-9/16//70-RTE 9 POTTERSVILLE-8/
112		
113	779	C4-SCHROON-1-YASH@
114	7795	242-9#7#70-WELLS HOUSE HOTEL-1B-9/16#70-RTE 9 POTTERSVILLE-8/
115		
116		
117		
118		
119	813	861-8#15#70-WHITES RETREAT-8#1B#2G#3C#M5-9#2#70-9#9#70-SOUTH SCHROON/
120		
121	813	861-8#15#70-WHITES RETREAT-8-1B#2G#3C#M5-9#2#70-9#9#70-SOUTH SCHROON/
122		
123		
124		
125		

07/05/73

DATA KEYPUNCHED FROM SAMPLE CODE SHEETS

PAGE 1

1 DATA KEYPUNCHED FROM SAMPLE CODE SHEETS
2 00001 A5-ONEIDA-1-YASH@@
3 00002 5288--1-8#21#70-8#26#70-SHADY VALLEY *
4 00004 5302----8#26#70-9#1#70-SYLVAN BEACH NY *
5 00006 5303----8#24#70-9#1#70-SYLVAN BEACH NY *
6 00007 5305----8#24#70-9#1#70-SYLVAN BEACH *
7 00008 5308----8#25#70-10#6#70-ONEIDA LAKE SYLVAN BEACH NY *
8 00009 5335----8#22#70-8#24#70-SYLVAN BEACH *
9 00010 5336----8#21#70-8#24#70-SYLVAN BEACH *
10 00012 A5-ONEIDA-2-YASH@@
11 00013 5288--1-49-M-M-3-2-2-4-2-1#3#4#12#13#14-1#
12 00014 5302--21--M-2-2-1-4-4-1#3#4#5#8#12#14#18-1#
13 00015 5303--31-F-M-1-2-1-3--1#3 TO 8#14-5 YR *
14 00016 5305--28-M-M-2-4-1-4-4-4#6#10#12#18-1#
15 00017 5308--42-M-M-1-1-1-1-4-99-2#
16 00018 5335--42-M-M-2-2-1-2-3#4-1#3#4#5#8#9#12#14-30 YR *
17 00019 5336--59-M-M-2-2-1-3-4-1#3#4#8#9#14-15 YR *
18 00022 A5-ONEIDA-3-YASH@@
19 00023 5288--1049#18-F#47@SHOPPING-M@18#4#
20 000234 5302--M@24#VACATION-F@21@VACATION#
21 00024 5303--1#33#4-M#31#5-F#11#4-F#10#4-F#8#4-F#6#4-M=5=SANDY BEACH, NEW BEAC
22 00025 H-V#4#5#
23 00026 5305--R@23#5#
24 00026 5308#
25 00028 5336--M@54#8#4-F#56#8#4#

A.4 Sample Punched Data

DATA FROM SAMPLE CODE SHEETS, AFTER RENUMBERING

- 1 DATA FROM SAMPLE CODE SHEETS AFTER RENUMBERING
- 2 A5305288011-----8#21#70-8#26#70-SHADY VALLEY #
- 3 A5305302011-----8#26#70-9#1#70-SYLVAN BEACH NY #
- 4 A5305303011-----8#24#70-9#1#70-SYLVAN BEACH NY #
- 5 A5305305011-----8#24#70-9#1#70-SYLVAN BEACH #
- 6 A5305308011-----8#25#70-10#5#70-ONEIDA LAKE SYLVAN BEACH NY #
- 7 A5305335011-----8#22#70-8#24#70-SYLVAN BEACH #
- 8 A5305336011-----8#21#70-8#24#70-SYLVAN BEACH #
- 9 A5305288021--49-M-M-3-2-2-4-2-1#3#4#12#13#18-1#
- 10 A5305302021--21-M-2-2-1-4-4-1#3#4#5#8#9#12#14-30 YR #
- 11 A5305303021--31-F-M-8-2-1-3--1#3 TO #14-5 YR #
- 12 A5305305021--28-M-M-2-2-4-1-4-4#6#10#12#18-1#
- 13 A5305308021--42-M-M-1-1-1-1-4-99-2#
- 14 A5305335031--42-M-M-2-2-1-2-3#4-1#3#4#5#8#9#12#14-30 YR #
- 15 A5305336031--59-M-M-2-2-1-3-4-1#3#4#8#9#14-15 YR #
- 16 A5305302031-M#24#VACATION-F#21#VACATION#
- 17 A5305303031-F#49#18-F#47#SHOPPING-V#18#4#
- 18 A5305303031-F#32#4-V#31#5-F#11#4-F#10#4-F#8#4-F#6#4-M#5#SANDY BEACH, NEW BEACH-M#
- 19 A5305303031#45#
- 20 A5305305031-F#23#5#
- 21 A5305308031#
- 22 A5305335031-M#42#8-F#37#3#
- 23 A5305336031-M#54#8-4-F#56#8#4#

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 1

A.6 System JCL

```
1 ***** JCL LISTING FOR THE EVQS SYSTEM
2
3
4
5
6
7
8
9
10
11
12 *** INITIAL SET-UP PROGRAMS ***
13
14
15
16
17
18
19
20
21
22 //EVQSBGM JOB (2143,123,1,1),ROTH,CLASS=C
23 // EXEC GENPROC F
24 //SYSIN DD DATA
25 ./ ADD NAME=EVQSCOR1
26 /**
27 //EVQSCOR1 PROC NUM=1,OUT=A
28 //LIST EXEC PGM=OBCORR,REGION=50K
29 //STEPLIB DD DSN=CL.LLIB.SRES14,DISP=SHR
30 //SYSPRINT DD SYSOUT=&OUT
31 //SYSIN DD DSN=ROTH.EVQSDATA(RAW&NUM),DISP=SHR
32 //EVQSOUT DD DSN=ROTH.EVQSDATA(RAW&NUM),DISP=MOD,DCB=DSORG=PS
33 /** DOES A QUICK LIST FOR OBVIOUS ERRORS
34 /** TO DO LIST WITHOUT WRITING ON EVQSOUT--DUMMY IT OUT
35 ./ ADD NAME=EVQSCOR2
36 /**
37 //EVQSCOR2 PROC NUM=1,LIST=0,OUT=A
38 /** CORRECTION STEP
39 //CORR EXEC PGM=IEBUPDTE,REGION=60K,TIME=1
40 //SYSPRINT DD SYSOUT=&OUT
41 //SYSUT1 DD DSN=ROTH.EVQSDATA,DISP=OLD
42 //SYSUT2 DD DSN=ROTH.EVQSDATA,DISP=OLD
43 /** FIRST COMMAND CARD IS
44 /** ./ CHANGE NAME=RAW&NUM,SEQFLD=028
45 /**
46 /** TO GET VRFY FULL LIST SET 'LIST=8'
47 //LIST EXEC PGM=VRFY,REGION=90K,TIME=3,COND=(&LIST,LE,CORR)
48 //STEPLIB DD DSN=CL.LLIB.SRES14
49 //SYSPRINT DD SYSOUT=&OUT
50 //SYSIN DD DSN=ROTH.EVQSDATA(RAW&NUM),DISP=SHR
51 //CONTROL DD DSN=ROTH.EVQSDATA(VRFYDATA)
52 ./ ADD NAME=EVQSNUM
53 /**
54 //EVQSNUM PROC OUT=A,NUM=1
55 /**
```

```

56 //RENUM EXEC PGM=RENUM,REGION=70K,TIME=4
57 //STEPLIB DD DSN=CL.LLIB,SRES14,DISP=SHR
58 //SYSPRINT DD DSN=ROTH.EVQSDATA(AMCNUM),DISP=SHR
59 //SYSIN DD SYSOUT=OUT
60 //SYSPUNCH DD DSN=ROTH.EVQSDATA(NUMDNUM),DISP=(MOD,KEEP),
61 // DCB=(DSORG=PS,
62 // LIST,NUMD) MEMBER FOR CHECK
63 //LIST EXEC PGM=IEBGENER,REGION=70K
64 //SYSPRINT DD SYSOUT=OUT
65 //SYSUT1 DD DSN=ROTH.EVQSDATA(NUMDNUM),DISP=SHR
66 //SYSUT2 DD SYSOUT=OUT,DCB=(BLKSIZE=80)
67 //SYSIN DD DUMMY
68 // ADD NAME=EVQSFINE
69 //**
70 //EVQSFINE PROC OUT=A,NUM=1
71 //**
72 //** INTERMEDIATE SORTING
73 EXEC PGM=IERRG00,REGION=64K,PARM='CORE=44000'
74 //SYSOUT DD
75 //SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
76 //SORTIN DD DSN=ROTH.EVQSDATA(NUMDNUM),DISP=SHR,
77 // DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200,DSORG=PS)
78 //SORTOUT DD DSN=ROTH.NUMDNUM,DISP=(NEW,PASS),
79 // SPACE=(TRK,(20,10)),UNIT=DISK,
80 // DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200,DSORG=PS)
81 //SORTWK01 DD UNIT=DISK,SPACE=(TRK,(10,10)),DISP=NEW
82 //SORTWK02 DD UNIT=DISK,SPACE=(TRK,(10,10)),DISP=NEW
83 //SORTWK03 DD UNIT=DISK,SPACE=(TRK,(10,10)),DISP=NEW
84 //SYSIN DD DSN=ROTH.EVQSDATA(SORTDATA),DISP=SHR
85 //** COPY STEP-CREATE TEMP. COPY OF FINAL FILE
86 //COPY EXEC PGM=IEBGENER,REGION=60K,TIME=1
87 //SYSPRINT DD SYSOUT=OUT
88 //SYSIN DD DUMMY
89 //SYSUT1 DD DSN=ROTH.EVQSFINE,DISP=SHR
90 //SYSUT2 DD DSN=ROTH.TCNUM,DISP=(NEW,PASS),
91 // SPACE=(TRK,(20,10)),UNIT=DISK,
92 // DCB=(BLKSIZE=3200,RECFM=FB,LRECL=80)
93 //** FINAL MERGE
94 EXEC PGM=IERRG00,REGION=64K,PARM='CORE=44000'
95 //SYSPRINT DD SYSOUT=OUT
96 //SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
97 //SORTIN01 DD DSN=ROTH.NUMDNUM,DISP=OLD
98 //SORTIN02 DD DSN=ROTH.TCNUM,DISP=(OLD,DELETE,KEEP)
99 //SORTOUT DD DSN=ROTH.EVQSFINE,DISP=OLD,
100 // DCB=(BLKSIZE=80,LRECL=80,RECFM=FB)
101 //SYSIN DD DSN=ROTH.EVQSDATA(MERGDATA),DISP=SHR
102 //**
103 //** SET-UP INITIAL FILES
104 //**
105 //** EXEC PGM=IEFBRL45
106 //DD1 DD DSN=ROTH.EVQSFINE,SPACE=(CYL,(6)),
107 // UNIT=2311,VOL=SER=231122,DISP=(NEW,CATLG),
108 // DCB=(RECFM=FB,BLKSIZE=80,LRECL=80)
109 //**
110 //DD2 DD DSN=ROTH.WK05,SPACE=(CYL,(4,2)),

```

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 3

```
111 // UNIT=2311,VOL=SER=231122,DISP=(NEW,CATLG),
112 // DCB=(RECFM=FB,LRECL=80,BLKSIZE=3600)
113 //DD3 DD DSN=ROTH.DOCMN,DISP=(NEW,CATLG),UNIT=2311,
114 // DCB=(BLKSIZE=3600,LRECL=80,RECFM=FB),VOL=SER=231122,
115 // SPACE=(TRK,(5,10,10))
116 // EXEC PGM=IEBGNER,REGION=64K
117 //* SPECIAL STEP TO PLACE 1 BLANK CARD IN EVQSFILE
118 //* NEEDED BECAUSE PSAM ALWAYS SKIPS THE FIRST CARD
119 //SYSPRINT DD SYSOUT=A
120 //SYSUT1 DD *
121
122 //SYSUT2 DD DSN=ROTH.EVQSFILE,DISP=SHR
123 //SYSIN DD DUMMY
124 //
```

125
126
127
128
129
130
131
132
133
134
135
136
137 ***** VARIOUS COMPILEATION JCL *****

138
139
140
141
142
143
144
145
146
147 *** FOR RENUM ***

```
148
149
150
151
152 //EVQSCRNM JOB (2143,123,1,1),ROTH,CLASS=C
153 // EXEC PL1LFCL,PARM='NEST,SORMGIN=(2,72,1)'
154 //SYSIN DD *
```

155 ***** PROGRAM DECK GOES HERE *****

```
156
157
158 //LKED.SYSLMOD DD DSN=CL.LLIB.SRES14(RENUM),UNIT=2314,SPACE=,DISP=OLD
159 *** THE NEXT 2 STMTS ARE ONLY NEEDED TO CORRECT A
160 PREVIOUS VERSION *****
```

```
161 //LKED.SYSIN DD *
162 NAME RENUM(R)
163 //
```

```
164
165
```

378

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 4

```
166
167
168
169
170
171
172     *** FOR OBCORR ***
173
174
175 //EVQSCOC JOB (2143,123,1,1),ROTH,CLASS=C
176 // EXEC ASMGCL
177 //SYSIN DD
178
179     **** PROGRAM DECK GOES HERE ****
180
181 //LKED.SYSLMOD DD DSN=CL.LLIB.SRES14(PREPNH),DISP=OLD,SPACE=,UNIT=2314
182     *** THE NEXT 2 STMTS ARE ONLY NEEDED TO CORRECT A
183     PREVIOUS VERSION ****
184 //LKED.SYSIN DD *
185     NAME OBCORR(R)
186 //
187
188
189
190
191
192     ** FOR PREPNH
193
194 //EVQSCPP JOB (2143,123,1,1),ROTH,CLASS=C
195 // EXEC ASMGCL
196 //SYSIN DD *
197
198     **** PROGRAM DECK GOES HERE ****
199
200 //LKED.SYSLMOD DD DSN=CL.LLIB.SRES14(OBCORR),DISP=OLD,SPACE=,UNIT=2314
201     *** THE NEXT 2 STMTS ARE ONLY NEEDED TO CORRECT A
202     PREVIOUS VERSION ****
203     NAME PREPNH(R)
204 //LKED.SYSIN DD *
205 //
206
207
208
209
210
211     *** FOR VRFY
212
213
214
215
216 //EVQSCVRY JOB (1101,123,2,3),ROTH,CLASS=C
217 // EXEC PL1LFCL,PARM=*NEST,SORMGIN=(2,72,1)
218 //LKED.SYSLMOD DD DSN=CL.LLIB.SRES14(VRFY),UNIT=2314,SPACE=,DISP=OLD
219 //SYSIN DD *
220
```

379

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 5

```
221          **** PROGRAM DECK GOES HERE
222 //
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238 **      COMPILER EVQS TERMINAL SYSTEM ASSEMBLER SECTIONS
239 **      IT ALSO GENERATES AN OBJECT DECK FOR BACKUP
240
241
242
243
244 //EVQTOTDK JOB (2143,123,5,6,500),ROTH,CLASS=C
245 // EXEC ASMD,PARM=BATCH,DECK,LREF,ESD,RLO,FULLXREF
246 //ASM.SYSLIB DD
247 //          DD DSN=ECOL.MACLIB,DISP=SHR
248 //SYSIN DD *
249
250 -          **** PROGRAM DECK GOES HERE
251
252
253
254 // EXEC ALPHLKED,USERID=SRES14,NAME=EVQS
255 //LOADMOD DD DSN=LOADSET,DISP=OLD
256
257     *** THE NEXT 6 CARDS ARE ONLY NEEDED TO PUT IN CORRECTIONS
258 //MYLIB DD DSN=CL.LLIB.SRES14,DISP=SHR
259 //SYSIN DD *
260     ENTRY EVQS
261     INCLUDE LOADMOD
262     INCLUDE MYLIB(EVQS)
263     NAME EVQS(R)
264 //
265
266
267
268     *** THE NEXT NINE CARDS ARE NEEDED FOR THE INITIAL COMPILATION
269 //***** CHECK WITH JOHN FISHER ABOUT FSAMLIB
270 //FSAMLIB DD DSN=ECOL.LINKLIB,DISP=SHR
271 //SYSIN DD *
272     ENTRY EVQS
273     INCLUDE LOADMOD
274     NAME EVQS
275     ENTRY FSAM
```

380

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 6

```
276 INCLUDE FSAMLIB(FSAM)
277 NAME FSAM
278 //
279
280
281
282
283
284
285
286
287
288 *** FOR FORTRAN PRINT ROUTINE EVQSRPT
289
290
291 //EVQSCPTF JOB (2143,123,1,1),ROTH,CLASS=C
292 // EXEC FORTGC
293
294 ** PROGRAM DECK GOES HERE
295
296 // EXEC ALPHLKED,USERID=SRES14,NAME=EVQS
297 //LOADMOD DD DSN=&LOADSET,DISP=OLD
298 //MYLIB DD DSN=CL.LLIB.SRES14,DISP=SHR
299 //SYSIN DD *
300 ENTRY EVQS
301 INCLUDE LOADMOD
302 INCLUDE MYLIB(EVQS)
303 NAME EVQSR)
304 //
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328 *** MAINTENANCE PROGRAMS
329
330
```

```

331
332
333
334      /**          JCL TO ADD TO THE DOCUMENTATION FILE
335
336
337      //EVQSOOC JOB (2143,123,1,1),ROTH,CLASS=C
338      // EXEC PGM=IEBUPDTE,REGION=64K
339      //SYSPRINT DD SYSOUT=A
340      //SYSUT1 DD DSN=ROTH.DOCMN,DISP=SHR
341      //SYSUT2 DD DSN=ROTH.DOCMN,DISP=SHR
342      //SYSIN DD *
343      ./ ADD NAME=SPALL
344      THE FOLLOWING SAMPLE QUERIES ARE ON THE SYSTEM
345      HELP      ITEM      FORM
346      -----
347      CDM5      1          A7
348      ./ ADD NAME=SPCOMS
349      VALID COMMANDS ARE:
350      ASSIGN,COPY,IF,QUERY,SCAN,TALLY,TRANSLATE
351      !! SEE TERMINAL USER'S MANUAL !!
352      ./ ADD NAME=ITEM1
353      ITEM ENTRIES WILL NOT BE USABLE TILL
354      !! JACK'S THESIS TABLE IS READY AND ENTERED !!
355      ./ ADD NAME=FORMAT
356      FORM ENTRIES WILL NOT BE USABLE TILL ENTERED FROM TABLE
357      -- IN THE MEAN TIME SEE QUESTIONAIRE PACKAGE.
358      //
359
360
361
362
363
364
365
366
367
368      ** TO RESTORE VERIFICATION DATA SETS
369
370
371
372      //EVQSRENW JOB (2143,123,2,3),ROTH,CLASS=C
373      // EXEC PGM=IEFBRL4
374      //DATA DD DSN=ROTH.EVQSODATA,DISP=(OLD,DELETE)
375      /**          SET-UP DATA SETS
376      //SETUP EXEC PGM=IEBUPDTE,REGION=64K,PARM=NEW,COVD=EVEN
377      //SYSPRINT DD SYSOUT=A
378      //SYSIN DD *
379      ./ ADD NAME=SORTDATA
380      SORT FIELDS=(1,11,CH,A),SIZE=E2500
381      ./ ADD NAME=MERGOATA
382      MERGE FIELDS=(1,11,CH,A)
383      ./ ADD NAME=VRFYDATA
384      TAB(2)=12 TAB(3)=11
385      ./ ENDUP

```

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 8

386 //SYSUT2 DD DSN=ROTH.EVQSDATA,SPACE=(CYL,(4,2,15)),
387 // UNIT=2311,VOL=SER=231122,DISP=(NEW,CATLG),
388 // DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)
389 //
390

391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416

417 *** JCL TO PREPNH CARDS IN BATCH
418 WITH PARM FOR LATEST BATCH,PUNCHED
419

420
421 //EVQS01 JOB (1101,123,5,1,4500),ROTH,CLASS=C
422 // EXEC PGM=PREPNH,SPACE=10K,PARM='40001,44000'
423 //STEPLIB DD DSN=CL.LLIB.SRES14,DISP=SHR
424 //FY07F001 DD SYSOUT=B,DCB=(RECFM=F,LRECL=80)
425 //

426
427
428
429
430
431
432
433
434
435
436
437
438
439
440

437 *** MISC. JOBS

383

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 9

```
441
442
443   ** JCL TO LIST EVQSDATA LIBRARY
444
445
446
447
448 //EVQSGEN2 JOB (1101,123,1,1),ROTH,REGION=60K
449 // EXEC PGM=IEBTPCH
450 // PRINT TYPE=PD,MAXFLOS=1,MAXNAME=1
451 // RECORD FIELD=(80)
452 //SYSUT1 DD DSN=ROTH.EVQSDATA,DISP=OLD,DSN=ROTH.EVQSDATA
453 //SYSUT2 DD SYSOUT=A
454 //
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472   ** JCL TO LIST MASTER DATA FILE
473
474 //EVQSCHK JOB (2143,123,1,2),ROTH,CLASS=C,REGION=40K
475 // EXEC PGM=DCSCOPY,REGION=40K
476 //STEPLIB DD DSN=CL.LLIB.OFC06,DISP=SHR
477 //SYSUT1 DD DSN=ROTH.EVQSFILE,DISP=SHR
478 //SYSUT2 DD SYSOUT=A,DCB=BLKSIZE=80
479 //
480
481
482
483
484
485
486
487 //EVQSRCVR JOB (2143,123,2,1),ROTH,CLASS=C
488 // EXEC PGM=IEBGENER,REGION=70K
489 //SYSPRINT DD SYSOUT=A
490 //SYSUT1 DD DSN=ROTH.TC11,DISP=(OLD,DELETE),VOL=SER=231403,UNIT=2314
491 //SYSUT2 DD DSN=ROTH.EVQSFILE,DISP=OLD
492 //SYSIN DD DUMMY
493 // EXEC PGM=IFBCOPY,REGION=70K
494 //SYSUT1 DD DSN=ROTH.EVQSDATA,DISP=OLD
495 //SYSUT2 DD DSN=ROTH.EVQSDATA,DISP=OLD
```

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 10

```
496 //SYSIN DD *
497 COPY OUTDD=SYSUT2, INDD=SYSUT1
498
499 *** DELETE BOTH RAW AND NUMD MEMEBRS IF WANT TO RE-ENTER DATA
500 EXCLUDE MEMBER=RAW11,NUMD11
501
502 *** ONLY DELETE NUMD MEMBER IF JUST WANT TO RENUM
503 EXCLUDE MEMBER=NUMD11
504 //
505
506
507
508
509
510 *** JOB TO LIST STATUS OF ALL DATA SETS - ALSO COMPRESS'S PDS'S
511
512
513
514 //EVQSLIST JOB (1101,123,1,3),ROTH,CLASS=C
515 // EXEC COMPRESS,DS='ROTH.EVQSDATA',REGION=60K
516 // EXEC COMPRESS,DS='CL.SLIB.SRES14',REGION=60K
517 // EXEC COMPRESS,DS='CL.LLIB.SRES14',REGION=60K
518 // EXEC PGM=IEHLIST,REGION=50K
519 //DD1 DD UNIT=2311,VOL=SER=231122,DISP=OLD
520 //DD2 DD UNIT=2314,VOL=SER=231402,DISP=SHR
521 //SYSPRINT DD SYSOUT=A
522 //SYSIN DD *
523 LISTVTOC FORMAT,VOL=2314=231402,DSNAME=CL.LLIB.SRES14
524 LISTVTOC FORMAT,VOL=2314=231402,DSNAME=CL.SLIB.SRES14
525 LISTPDS FORMAT,VOL=2314=231402,DSNAME=CL.LLIB.SRES14
526 LISTPDS FORMAT,VOL=2311=231122,DSNAME=ROTH.EVQSDATA
527 LISTVTOC FORMAT,VOL=2311=231122
528 // EXEC PGM=IEBPTPCH,REGION=64K
529 //SYSPRINT DD SYSOUT=A
530 //SYSUT1 DD DSN=ROTH.EVQSDATA,DISP=SHR
531 //SYSUT2 DD SYSOUT=A
532 //SYSIN DD *
533 PRINT TYPORG=PO,MAXFLDS=1
534 RECORD FIELD=(80,,25)
535 //
536
537
538
539
540
541
542
543 *** BATCH JCL WITH SOME SAMPLE CONTROL CARDS
544
545
546
547
548 //EVQSRUN JOB (1101,123,1,1),ROTH,CLASS=C
549 //EVQSGO EXEC FORTGLOD,REGION=64K,PARM='EP=EVQS'
550 //STEPLIB DD DSN=CL.LLIB.SRES14,DISP=SHR
```

385

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 11

```
551 // DD DSN=SYS1.FORTLIB,DISP=SHR
552 //SYSLTB DD DSN=CL.LLIB.SRES14,DISP=SHR
553 // DD DSN=SYS1.FORTLIB,DISP=SHR
554 //SYSLOUT DD SYSOUT=A
555 //SYSLIN DD DSN=CL.LLIB.SRES14(EVQS),DISP=SHR
556 //* OUTPUT FILES
557 //SYSPRINT DD SYSOUT=A
558 //MASPRINT DD SYSOUT=A
559 //FTL6F001 DD SYSOUT=A
560 //SYSUDUMP DD SYSOUT=A
561 //* INPUT FILES ARE SYSIN AND ERRIN
562 //* WORK FILES
563 //EVQSWK01 DD DSN=ROTH.EVQSFILE,DISP=SHR
564 //EVQSWK02 DD DSN=EWK02,DISP=(,PASS),SPACE=(TRK,(10,10)),UNIT=DISK
565 //EVQSWK03 DD DSN=EWK03,DISP=(,PASS),SPACE=(TRK,(10,10)),UNIT=DISK
566 //EVQSWK04 DD DSN=EWK04,DISP=(,PASS),SPACE=(TRK,(10,10)),UNIT=DISK
567 //EVQSWK05 DD DSN=ROTH.WK05,DISP=OLD
568 //* MISC FILES
569 //DD1 DD VOL=SER=231122,UNIT=2311,DISP=SHR
570 //DD2 DD VOL=SER=231401,DISP=SHR,UNIT=2314
571 //SYSIN DD *
572 ASSIGN OUTPUT=TERMINAL
573 SCAN RESP=(01-26)
574 ASSIGN OUT=3,IN=3
575 SCAN FORM=A1,LK=2,RESP=(01-26)
576 END
577 ASSIGN OUT=PRINT
578 SCAN FORM=A2,RESP=(03-05)
579 SCAN LK=3,RESP=(01-26)
580 ASSIGN
581 //
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
```

***** EXAMPLES OF VERIFICATION PROCESS *****

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 12

```
506
507
508
509
510
511
512   /** JOB TO GENERATE JOBCORR LISTING AND ENTER DATA
513   //EVQSR11A JOB (2143,123,1,2),ROTH,CLASS=C
514   /**ROUTE! PRINT PRINTER! CARD GOES HERE
515   // EXEC EVQSCOR1,NUM=11
516   //SYSIN DD *
517
518   ** BOX OF RAW DATA CARDS GOES HERE.
519   *** STARTING WITH A HEADER
520   AND ENDING JUST BEFORE A MUCH LATER HEADER CARD
521   //
522
523
524
525
526
527
528
529
530
531
532   ** JOB TO CORRECT AND RENUMBER RAW DATA
533
534   *** CORRECTION CARDS ARE THE MAJORITY OF THOSE FOR
535   SCHROON LAKE BOX 1
536
537   //EVQSRN11 JOB (1101,123,3,3),ROTH,CLASS=C
538   // EXEC EVQSCOR2,NUM=11
539   //SYSIN DD *
540   /* CHANGE NAME=RAW11,SEQFLD=028
541   52 147-LIVINGSTON NJ ESSEX-TRANSPORTATION ENGINEER-15-1--2-13#15#
542   75 173--42-M-M-4-2#3-1-4-1-4#5#6#8#12#14#15-3#
543   101 174-7#5#70-EAGLE PT CAMPSITE-3A-1B-6#30#70 TO 7#6#70-7#9#70-EAGLE POI
544   101 LOONT #
545   106 177--60-M F-M-10-2-1-4--1 TO 6#12#14#15#99-3#
546   124 174-PLYMPTON MASS. PLYMOUTH-TRUCK DRIVER-20-1--2-99#
547   187 964--58-M-M-5-2-1-4-1-4#5#8#15-4YR#
548   188 966--48-F-M-3-2-1-2-1-4#12#15#99-2YR#99#
549   190 975--38-M-M-4-2-1-4#99-1-3#4#12#15-3YR#
550   193 986--45-M-M-4-2-1-4#99-1-4#5#15-4YR#
551   292 A5-SCHROON-3-YASHO0
552   365
553   381
554   382 5045-M#31#4#GOLF-F#30#4#6-F#7,5#4#6-F#3,5#4#
555   510 LK ESSE#
556   529 567-1-1#12#99-2-2--150-1-17 05 08 #
557   530 566-1-1#4-2-2--200-1-17 75 08#12 10 08 #
558   543 571-25-1-1956-2000--1-2-1-2-9#
559   567 2034#G1#3067-9#7#70-W SIDE SCHROON LK-1C#2E-9#10#70-9#16#70-S SCHROON
560   573
```

07/05/73

JCL LISTING FOR THE EVQS SYSTEM

PAGE 13

```

661      575      2043#G1#3076#NR-9#7-70-W SIDE SCHROON LK-1C#2E-9#8#70-9#11#70-S SCHRO
662      576      ON ESSE#
663      618      2045#G1#3078-9#7#70-W SIDE SCHROON LK-1C#2E-9#7#70-10#6#70-W SHORE SC
664      632      2061#G1#3094-9#7#70-W SIDE SCHROON LK-1C#2E-9#7#70-9#9#70-GROVE PT RD
665      641      2049-13-1-53-M-M-1-VALLEY STREAM NY MASS-5-2#
666      643      2038-13-1-60-F-S-1-BROCKLYN NY KING-6-1#
667      648      2045-1-1-1#99-2--125+1-60#
668      661      2062-75-1-1953-12000-13-3-1-2-90120#99#
669      697      2108-13-1-36-M-M-1--6-6#
670      700      2084-99-3--2#99--10-1-16 120 10 #
671      703      2092-1-3---125#99-1-14 40 DB# DB#
672      7645     203-2-1968--NONE-2-----2-2#
673      779      SCHROON-I-YASH@#
674      7795     242-9#7#70-WELLS HOUSE HOTEL-1B-9#16#70-RTE 9 POTTERSVILLE-8#
675      813      861-8#15#70-WHITES RETRFAT-8-1B#2G#3C#M5-9#2#70-9#9#70-SOJTH SCHROON#
676      826      1722-36-M-M-2 TO 3-2#3-4#5#8#12#14-2-15 65 DB#15 7.5 DB---3#
677      857      1736-9#17#99-CANTON NY STLA-3-PROFESSIONAL POLICEMAN-7#99#
678      1160     ROOM#
679      1217     CABINS SCHROON LK#
680      1250
681      1627
682      // EXEC EVQSNM,NUM=11
683      //
684
685
686
687
688
689
690
691      *** DO FINAL SORT/MERGE
692
693
694
695
696      //EVQMRGE JOB (2143,123,1,1),ROTH,CLASS=C
697      // EXEC EVOSFINE,NUM=11
698      //
699
700
701
702
703
704
705

```

October 9, 1972

Q3 -- 'For each participant'

A --3's

Gen. -- for ranges of responses--

12 to 23 ⇒ 12 ∅ TO ∅ 23

12 → 23

Q16 Husband and wife--two parts of same question.

EVQS October 16, 1972

If out-of-state -- type whole county.

If in state -- First four letter of county name.

Abbreviations

Rd -- Road

RTE -- Route

N,S,E,W -- North, South, East, West

Lake -- LK

October 26, 1972

G1's & G2's

Page TYPE 1 HDR card

has --G1#B4--

↑
in first question

Each line has -- resp. no -- other resp. no. --

↑
G1

↑
B4

Other page types --eliminate column with other form type.

October 26, 1972

F2 Page 2 *

Question after sex is M.S.

} 2 separate question.

Next one is # people -- 1

(--between)

Where *

& abbreviation is used leave no space between

lloz

no space

<u>Sequence start</u>	<u>Lake</u>	<u>Form Type</u>
1 - 128	Schroon	A3
129 - 398	Schroon	A5
399 - 732	Schroon	B4

October 26, 1972

F 2 & B4

Boats --Q5 also

FT HP. I/OB with no special symbols unless missing parts then use

abbreviation after category present

FT

HP

(99 - for type)

Q16 -- F2 --

Fish species

EOQ between --species--no.#--size--WT--

W/IN each EOP

If missing corresponding one use multiple EOP

X X X
X NR X
X X NR

-X#X#X--X##X--X#XEOL

November 13, 1972

Q8 A5

If respondent is 1 or 2 put

1

2

If is no. of years put

1 yr

10 yr (yr=years with no space between)

November 27, 1972

Wherever a appears use CK

November 28, 1972

B-4 & A-7's

528
G1-1028



528#G1#1028#NR-

Corresponding G1 form

If N.R.

December 4, 1972

Form B5, Q136

Type boat number of (if none, just checked)



2(3) 3 of type 2

G2 Q1

Code book has b & c.

#(##)
↓ ↘
boat type seq. of name Punch as is--separate as separate parts

5(2)

8(3)

2(1)

If more than one name put parns after #

1(1,2)

February 13, 1973

1183-B4#683-

B4 Type Correction

LITERATURE CITED

The basic topic of this work, that of limited format data-formatting is a rare topic. As such, no works were found which dealt directly with the topic. Hence, the publications listed relate only to the various parts of the thesis; the general field of sociological data; the details of the Environmental Quality Survey; or, the technical aspects of the computer implementation, both design and specific machine formatting.

1. Lewis, Rosenkrantz, Stearns; Theory of Compiler Design, manuscript for R.P.I. Systems Division course of same name, 1972.
2. Bisco, R.L., ed., Data Bases, Computers & The Social Sciences; John Wiley & Sons, N.Y., 1970.
3. Roth, R., Preliminary Report on Computerizing Jack Kooyoomjian's Survey (EVQS); FWI Local Manuscript, September 1, 1972.
4. Struble, G., Assembler Language Programming For the IBM System/360; Addison-Wesley; Menlo Park, California, 1969.
5. IBM 360/50 OS/MVT Release 21 System Manuals: including:
 - 5.1 "Fortran Functions & Service Subprograms"
 - 5.2 "System Guide to Debugging"
 - 5.3 "Assembler G Language Manual"
 - 5.4 "Assembler G Programmer's Guide"
 - 5.5 "System Control Blocks Manual"
 - 5.6 "System I/O MACRO Facilities"
6. Various R.P.I. Office of Computer Services Publications, including:
 - 6.1 Alpha Time-Sharing System Terminal User's Manual
 - 6.2 Assorted Internal FWI & FWI Computer Group Publications, Notes, Correspondence & Pre-publication Papers by/with K. Jack Kooyoomjian.