APPENDIX E

SOLUTION TO CHAPTER SELF-TEST

CHAPTER 1

TRUE-FALSE

1. T
2. F— The AS/400 family of computers, as with all IBM midrange and mainframe computers, uses the EBCDIC coding system.
3. F—Arrival sequence files do not have assigned key fields. Thus, they are accessed in a first-in, first-out sequence.
4. T
5. T
6. F—The length of numeric fields on the DDS indicates the number of digits, not the number of bytes in the field. Thus, a length of 9 indicates that the numeric field will hold 9 digits and occupy 4 bytes of storage.
7. F—The `DSPFFD` command is used to display the description of the file and attributes of the record format and fields contained in the record.
8. F—The `DSPPFM` command is used to display the data contents of the records in a file.
9. T
10. T

FILL-IN-THE-BLANKS

1. data; index or key
2. key
3. key; pointer
4. related; joined
5. relational; DB2/400
6. arrival-sequence
7. keyed
8. physical
9. Data File Utility (DFU)
10. file; record; field; key
11. Display Field File Description
12. Display Physical File Member
13. Copy File
14. rows
15. sequentially; randomly

CHAPTER 2

TRUE-FALSE

1. F—The compiler does not detect logic errors
2. F—All symbolic programs need to be compiled in order to create an object program ready for execution.
3. F—The `PROCEDURE DIVISION` is usually the longest division because it is where all of the processing is done.
4. T
5. F—File-names must not contain embedded blanks.
6. T
7. T
8. T
9. T
10. F—As many files as are needed may be defined.

FILL-IN-THE-BLANKS

1. identify
2. assign; device
3. input; output; work areas
4. instructions; operations
5. `OPEN`
6. `PERFORM...UNTIL`
7. `8, 9, 10, 11`
   `12, 13, 14,` and so on
8. `IDENTIFICATION; ENVIRONMENT; DATA; PROCEDURE`
9. B
10. `DIVISION; SECTION`
11. period; space or blank
12. `PROGRAM-ID.`
13. `CONFIGURATION; INPUT-OUTPUT`
14. `FILE-CONTROL`
15. `B`

CHAPTER 3

TRUE-FALSE

1. T
2. T
3. T
4. T
5. F—ARE-THERE-MORE-RECORDS would be defined in the WORKING-STORAGE SECTION
6. F—The last statement in should be a READ.
7. T
8. T
9. T
10. T
11. T
12. F—Fields defined in a record description must appear in the same order in which they are sequenced in the disk record.
13. F—Valid subordinate level numbers range from 02 to 49.
14. T
15. F—Record names are assigned in the DATA DIVISION.

FILL-IN-THE-BLANKS

1. SELECT; FD; OPEN, CLOSE, and READ
2. AT END
3. filename; record name
4. PERFORM
5. is not
6. modules, routines or paragraphs
   sentences or instructions or statements
7. opened
8. sixteenth
9. FILE; WORKING-STORAGE
10. thirty; alphabetic; special; embedded blanks
11. must
12. FD
13. 01
14. 02; 49
15. 02; 01
16. A; B
17. DATA
18. elementary
19. type; size
20. X's; A's; 9's
21. characters or bytes
22. 2-position
23. V999
24. WORKING-STORAGE
25. VALUE

CHAPTER 4

TRUE-FALSE

1. F—The VALUE clause sets the initial value for a field. During execution, new data can be placed into any field that was defined using the VALUE clause.
2. F—The field will contain XYZ
3. T
4. F—Moved as 092␣170
5. F—VALUE clauses are used in the WORKING-STORAGE SECTION to establish an initial value.
6. T
7. F—Two floating-string characters may not be used in one report-item. If a dollar sign is to float, then a sign may not be placed in the leftmost position of the field. However, signs may appear in the rightmost position of a report-item. Thus, $$,$$$.99 - is valid.
8. T
9. F—The VALUE clause is used only with those fields in the WORKING-STORAGE SECTION that require an initial value.
10. T

FILL-IN-THE-BLANKS

1. report; column
2. VALUE
3. 9
4. XBXB(13)
5.

```
CA   B   1   2   2   2   3   3   4   4   4   5   5   6   6   6   7
78901234567890123456789012345678901234567890123456789012345678901234567890123456789012
     WRITE PRINT-RECORD FROM HEADING-RECORD
         AFTER ADVANCING 2 LINES.
```

6.

```
CA   B   1   2   2   2   3   3   4   4   4   5   5   6   6   6   7
78901234567890123456789012345678901234567890123456789012345678901234567890123456789012
     WRITE PRINT-RECORD FROM HEADING-RECORD
         AFTER ADVANCING PAGE.
```

7. ACCEPT
8. report item with edit symbol(s) specified in the PICTURE clause.
9. Z
10. QUOTE

CHAPTER 5

TRUE-FALSE

1. T
2. F—A flowchart is not very useful as a planning tool if it is drawn after a program has been coded.
3. F—It may have logic errors as well.
4. F—"Top-down" refers to the hierarchical representation of modules; "structured" refers to the fact that a program uses the modular approach.
5. T—Routine is also a synonym.
6. T—Flowcharts and pseudocode are language-independent.
7. F—A hierarchy chart illustrates the relationships among modules.
8. T
9. T
10. F—It might be sufficient, but it is important to test all possible situations that could arise in a program.

FILL-IN-THE-BLANKS

1. pseudocode (Flowcharts were not originally developed for structured programs.)
2. hierarchy or structure chart
3. sequence
4. `PERFORM...UNTIL`
5. a predefined process
6. numbering modules to help locate them in a large program (`100-`, `200-`, etc.); describing the nature of the module (`ERROR-ROUTINE`, `TOTAL-ROUTINE`, etc.)
7. `IF; ENDIF`
8. structure chart
9. logic
10. operations or functions
11. a parallelogram
12. a rectangle
13. functions
14. selection
15. `ENDIF`

CHAPTER 6

TRUE-FALSE

1. T
2. F—The figurative constant `ZEROS` may be moved to alphanumeric or numeric fields.
3. F—`FILLER` is an old COBOL reserved word used to designate a field that will not be accessed by the program; it cannot be used in the `PROCEDURE DIVISION`.
4. F—A blank in a numeric field can cause a program to abort.
5. F—Data in an alphanumeric field is left-justified. Data in a numeric field is right-justified and zero-filled without changing its numeric value (e.g., `123` in a five-position field is represented as `00123`).
6. T
7. F—Low-order nonfilled positions are replaced with spaces.
8. T
9. T
10. F—High-order truncation is the result of a receiving field being shorter than the sending field.

FILL-IN-THE-BLANKS

1. verb or operation.
2. sending field.
3. receiving field
4. `SAM`; `SAM` (*Note*: contents of a sending field remains unchanged.)
5. identifier; quotation marks
6. `453; 453`
7. alphanumeric
8. enclosed in quotation marks
9. `XXX` or `X(3)`
10. is not (Literals appearing in the `PROCEDURE DIVISION` need not be defined elsewhere in the program.)
11. numeric; nonnumeric (or alphanumeric) literals
12. identifier or data-name
13. figurative constant; alphanumeric or alphabetic; blanks or spaces
14. `99V999`
15. `0ΔΔΔΔ` (Δ denotes a blank); would not be

CHAPTER 7

TRUE-FALSE

1. F—`SUBTRACT A FROM 150` is not valid. The receiving field, which is the one holding the result, must be a data-name, not a literal.
2. T
3. T
4. T
5. T

FILL-IN-THE-BLANKS

1. `29`
2. `19`
3. `NET`; it remains unchanged; `7000` (The original `2000` in `NET` does not enter into the calculation.)
4. identifier
5. precedes

CHAPTER 8

TRUE-FALSE

1. T
2. T
3. F—Zero is neither negative nor positive. An alphanumeric field can be neither numeric nor alphabetic.
4. F—In EBCDIC, numbers are greater than uppercase letters. In ASCII however, numbers are less than uppercase letters.
5. F—Numeric items are compared algebraically (020 = 20); trailing spaces do not affect the evaluation of a nonnumeric field. (Expansion of the smaller field according to rules of numeric and alphanumeric fields results in fields of equal size for the comparison.)
6. T
7. T
8. T
9. T
10. F—Comparisons are made from left to right. ΔABC is not equal to ABCΔ

FILL-IN-THE-BLANKS

1. NOT
2. AND
3. conditional
4. ELSE
5. >=
6. EBCDIC
7. nested
8. SIGN; CLASS
9. condition-name
10. EVALUATE

CHAPTER 9

TRUE-FALSE

1. F—After the named paragraph(s) have been performed, control returns to the next instruction in sequence following the `PERFORM` statement.
2. T
3. T
4. T
5. T
6. T
7. T
8. F—The `GO TO` is an unconditional branch that permanently transfers control to another paragraph. The `PERFORM` is a temporary transfer of control that will return to the statement following the `PERFORM`.
9. T
10. T

FILL-IN-THE-BLANKS

1. the statement directly following the `PERFORM`.
2. no (0) times.
3. looping
4. Iteration
5. loop
6. infinite loop
7. nested
8. will not be
9. integer, identifier or arithmetic expression
10. `PERFORM...TIMES`

CHAPTER 10

TRUE-FALSE

1. T
2. F—The COPY statement retrieves and includes into a source program a series of prewritten COBOL entries. The CALL statement causes a separate subprogram to be executed.
3. T
4. T
5. F—In addition to the PROCEDURE DIVISION, the COPY statement may be used in the DATA and ENVIRONMENT DIVISIONs.
6. F—Only the order, size, and type of the variables are important, not the names.
7. F—If no data is passed, no LINKAGE SECTION is required.
8. F—The called program ends with an EXIT PROGRAM entry.
9. T
10. T

FILL-IN-THE-BLANKS

1. library; copying
2. COPY
3. make coding and debugging easier; increase standardization.
4. subprograms being called can be coded and executed as independent programs.
5. calling; called
6.

```
CA   B   1   2   2   2   3   3   4   4   4   5   5   6   6   6   7
78901234567890123456789012345678901234567890123456789012345678901234567890123456789012
      CALL 'literal-1'
           USING identifier-1.
```

7. calling
8. LINKAGE
9. USING
10. EXIT PROGRAM

CHAPTER 11

TRUE-FALSE

1. F—Logical files provide an access path to data stored in physical files.
2. T
3. T
4. F—Many logical files do not have unique keys because they provide an access path to data that requires keyed access by fields that contain duplicate values.
5. F—Logical files containing multiple-record formats contain a record format for each file specified in the logical file.

FILL-IN-THE-BLANKS

1. simple
2. multiple-record format
3. access path
4. `PFILE`
5. `JFLD`
6. `JDFTVAL`
7. `JFILE`
8. `JREF`
9. `VALUES`
10. `JOIN`

CHAPTER 12

TRUE-FALSE

1. F—Sorting is by minor fields within major fields.
2. T
3. T
4. T
5. F—The hold field should be reinitialized to the value contained in the control field of the current input record.
6. F—Some detail modules do not produce any output. That is, group printing alone may be sufficient.
7. T—Generally, only the level totals for the corresponding level break (and lower levels) are set to zero. Higher level totals are not reset at lower level breaks.
8. T
9. F—Any number of levels of control breaks is permitted.
10. T

FILL-IN-THE-BLANKS

1. a hold field
2. control break
3. the control total; the control field; control field to the hold field
4. detail processing or batch processing
5. a minor-level control break

CHAPTER 13

TRUE-FALSE

1. T
2. F—The `START` positions the file at the correct location but it does not read a record.
3. F—Only if the file has been accessed dynamically or an `ALTERNATE RECORD KEY` is being used for sequential access.
4. T
5. T
6. T
7. T
8. T
9. T
10. T.

FILL-IN-THE-BLANKS

1. `RECORD KEY`
2. `DELETE` file-name
3. `RECORD`
4. `WRITE`
5. `INVALID KEY`
6. `I-O`
7. `DYNAMIC`
8. `REWRITE`
9. `START`
10. `DYNAMIC`

CHAPTER 14

TRUE-FALSE

1. T
2. T
3. F—Fields may be defined as input, output or both.
4. F—The best format for entering data is the data entry format.
5. T
6. T
7. F—Indicators can be from 1 to 99.
8. F—`EDTCDE` is used for most editing functions. `EDTWRD` is used to provide editing functions not available with the `EDTCDE` keyword.
9. F—`DSPATR` is used to define the attributes of an element in the record format.
10. T

FILL-IN-THE-BLANKS

1. batch
2. off-line
3. on-line or interactive
4. on-line
5. on-line or interactive
6. `WORKSTATION` file
7. prompts
8. query or informational
9. menu
10. `OVERLAY`

CHAPTER 15

TRUE-FALSE

1. T
2. T
3. F—It is the name specified on the `OCCURS` level.
4. F—The index must be initialized with a `SET` statement prior to the `SEARCH`.
5. F—The `SEARCH ALL` statement is used to perform a binary search.
6. T—The index is automatically set at the appropriate point when a binary search is performed.
7. T
8. F—If the field is a numeric integer it may be used as a subscript.
9. T
10. T
11. T
12. T
13. F—Indexes are initialized with a `SET` statement.
14. F—A serial search must be used when the entries in the table either are not in a particular sequence or are organized so that the first entries are the ones encountered most frequently. A binary search, however, is the most efficient method of accessing the table when entries are in order.
15. T

FILL-IN-THE-BLANKS

1. first; middle
2. `ASCENDING` or `DESCENDING KEY`
3. `WED`
4. index
5. (a) all amounts have the same format or `PIC` clause
   (b) subscript

CHAPTER 16

TRUE-FALSE

1. F—The first subscript from the left refers to the major-level `OCCURS`.
2. F—COBOL 85 permits seven levels of `OCCURS`.
3. T
4. T
5. F—An entry in a table is referenced by the lowest-level data-name. In this case, `ITEM-2` must be used.

FILL-IN-THE-BLANKS

1. major; minor
2. (a) 2000 (20 x 50 x 2)
   (b) `ITEMX` (or `FIELDXX`)
   (c) 1 to 20; 1 to 50
3. major; intermediate; minor
4. comma; space
5. `INDEXED BY`

CHAPTER 17

TRUE-FALSE

1. T
2. F—The subfile control-record format can contain fields as well as headings.
3. F—The SFLDSP (Subfile Display) keyword indicates when to display the subfile.
4. T
5. T

FILL-IN-THE-BLANKS

1. `INDARA`
2. `SFLCTL`
3. `SFLSIZ`
4. `SFLPAG`
5. `SFLCLR`

CHAPTER 19

TRUE-FALSE

1. T
2. F—spacing
3. F—`DATE` is the field that contains the system date.
4. T
5. T

FILL-IN-THE-BLANKS

1. externally described
2. Data Description Specifications
3. skipping
4. `SKIPB; SKIPA`
5. `FILE SECTION; DATA`