Eleanor Godfrey:
Honors House

Dear Dr. Godfrey:

Here is my finished capstone. I had a lot of fun doing it. This is the only copy of my project I have, so I would appreciate it if I could have it back when you are done reviewing it. No hurry though.

Thank you.

Sincerely,
Jean Hoch
I feel that my Senior Capstone was an appropriate project to finish my undergraduate college education with. During the current semester I worked with Kathi Davis on supplementary material for a PL/1 textbook. PL/1 is a computer programming language similar to both FORTRAN and COBOL. Kathi Davis, along with Lyle Domina, are presently writing *Structured Programming: PL/1 with PL/C*. My task was to write exercises and problems for chapters one through seven along with the answers. After I completed the above work, I wrote sample test questions for each chapter to be included in a teacher's aid workbook.

Having never done this type of writing before, I needed to set some goals for myself. The most important goal was to write exercises that would completely test the readers' knowledge of the textbook material. I hope those students who work out the chapter exercises and problems will learn more about PL/1 and also realize what areas of the language they need to study more extensively.

Another important point I always had to remember is that my programming, flowcharting and pseudocoding techniques should be consistent with those of the authors'. It would be very confusing for a student using the PL/1 textbook to try to figure out which is correct: the text or the answers to the exercises. One way I tried to accomplish this consistency was to test all the answers for exercises and problems for correctness using the computer and the PL/C compiler. This is the compiler used by the authors.

I learned a great deal while working on my Senior Capstone. In addition to learning the importance of my work being consistent with the authors', I received some first hand knowledge in what is entailed when writing a textbook. One of the very first steps is setting goals, such as what I want my readers to gain from my writing. Outlines are also very important when drawing up a chapter of exercises and problems. I found outlining the chapter was beneficial in helping me write questions that cover all important information. After all the work is done, there are editors and PL/1 instructors from other universities who read the textbook and suggest improvements.

Overall this Senior Capstone was a positive experience and allowed me to use the knowledge I have gained in the past four years as a computer science major. The remaining portion of this report contains the exercises, problems, answers, and test questions I have created for chapters one through seven of *Structured Programming: PL/1 with PL/C*. 
CHAPTER 1

EXERCISES

1. What value is printed in each of the algorithms presented in the following flowcharts?

2. Write the pseudo-code represented by the flowcharts in Exercise 1.

3. Draw the flowchart corresponding to each of the following algorithms.

   A. START
      Read X,Y
      Write X,Y
      IF (X < Y)
         Z <-- Y - Z
      ELSE
         Z <-- X - Y
      ENDIF
      Write Z
      END

   B. START
      Read N
      CTR <-- 0
      PCTR <-- 0
      DO WHILE (CTR < N)
         Read X
         IF (X < 0)
            PCTR <-- PCTR + 1
         ENDIF
         CTR <-- CTR + 1
      ENDDO
      Write CTR, PCTR
      END

   C. START
      Read A,B
      Print A,B
      C <-- 2 * B + A - 3
      Write C
      END

PROBLEMS

1. Write the pseudo-code and draw the flowchart for each of the following programs.

   A. Read, print and calculate the sum of 100 numbers. Print the sum when finished.

   B. Read ten numbers, and compute the average of those numbers that are greater than zero. Print the average.

   C. Read five numbers, and print the maximum value.
CHAPTER 2

EXERCISES

1. A. How many halfwords are in two doublewords? How many bytes? How many bits?
   
   B. What number system is used by the computer?
   
   C. List and describe the two stages that are involved in the processing of a PL/1 program.

2. Identify the invalid labels and give the reason why they are not valid.
   
   A. ABC_DEF_GHI_JKL_MNO_PQR_STU_VWX_YZ;
   
   B. aPRICE;
   
   C. SUM_OF_NUMBERS;
   
   D. 260B_PL1;
   
   E. CODE 3;
   
   F. TOTALS;

3. What would VALUE1 contain after each of the following instructions?
   
   A. VALUE1 = 4 + 6 / 2;
   
   B. VALUE1 = (3 - 4) * 6;
   
   C. VALUE1 = -6 / 2;
   
   D. VALUE1 = (4 * 4) * 3;

4. Show the output of the following programs.

   A. /* FIGURE NET PAY */
      PAYROLL: PROCEDURE OPTIONS (MAIN);
      DECLARE WAGES;
      DECLARE TAX;
      DECLARE PAY;
      WAGES = 200;
      TAX = 20;
      PAY = WAGES - TAX;
      PUT DATA;
      END PAYROLL;
B. /* COMPUTE THE AVERAGE OF THREE NUMBERS */
AVERAGE: PROCEDURE OPTIONS (MAIN);
DECLARE NUM1;
DECLARE NUM2;
DECLARE NUM3;
DECLARE AVE;

NUM1 = 4;
NUM2 = NUM1 * 10;
NUM3 = NUM1 + NUM2 - 2;
AVE = (NUM1 + NUM2 + NUM3) / 3;

PUT DATA;
END AVERAGE;

C. /* THE NUMBER OF SECONDS IN A DAY */
TIME: PROCEDURE OPTIONS (MAIN);
DECLARE HOURS;
DECLARE MINUTES;
DECLARE SECONDS;

HOURS = 24;
MINUTES = HOURS * 60;
SECONDS = MINUTES * 60;

PUT DATA;
END TIME;

PROBLEMS

1. Write a PL/1 program to add the numbers 245 and 73 and print the result.

2. To graduate from State University a student must complete 125 credit hours. Debbie enters State University with 20 credit hours and plans to take 15 hours per semester. Write a program to compute how many semesters Debbie should complete to qualify for graduation.

3. If an accounting student received an A in Auditing, C in Calculus, and a B in English, write a program to compute the student's grade point average. (Assume A = 4.0, B = 3.0, C = 2.0, and Auditing = 4 hours, Calculus = 3 hours, English = 2 hours).
CHAPTER 3

EXERCISES

1. List the four main attributes of numeric data in PL/1 and the options of each possible attribute.

2. Show the statements necessary to define the following variables and initialize them as indicated.
   
   A. SUM - an integer of 6 digits initialized to zero.
   
   B. AVERAGE - a number of 4 digits left of the decimal point and 2 digits right of the decimal point.
   
   C. INTEREST - a scientific number with a precision of 8 and initialized to 1.
   
   D. COUNTER - a binary number of 15 bits initialized to 1.

3. Evaluate the following expressions given that $A = 5$, $B = 3$, $C = 1$, and $D = 8$. Keep in mind the order of arithmetic expression rules.
   
   A. $2 + A ** B / C - D * 10 + 4$.
   
   B. $C * D - 4 ** 2 + B / C - 9$.
   
   C. $25 * B - (A + B) / 2 * 8 - 12$.
   
   D. $((B + C) - 4) + 8) ** ((D - A) * (A - 5))$.
   
   E. $-2 ** 3$.

4. Given the following DECLARE statements, what would the PUT DATA; instruction print after each set of assignment statements?

   DECLARE
   NUMBER1 FIXED DECIMAL (8,4),
   NUMBER2 FIXED;

   A. NUMBER1 = 345;
      NUMBER2 = 123.45;
      PUT DATA;

   B. NUMBER1 = 0.02468;
      NUMBER2 = 3456789;
      PUT DATA;

   C. NUMBER1 = 123456.00
      NUMBER2 = 2.3456;
      PUT DATA;

   D. NUMBER1 = 123.456;
      NUMBER2 = 12340;
      PUT DATA;

   E. NUMBER1 = 12345.6;
      NUMBER2 = 000.123;
      PUT DATA;

   F. NUMBER1 = .123;
      NUMBER2 = 9876.5432;
      PUT DATA;
PROBLEMS

1. Write a program that will read in 3 prices of new cars and print the maximum of the prices. Assume the car prices are no greater than $9999.99. (Hint: Use a built-in function).

Read in 3 fixed integers that represent the sides of a box. Print these numbers. Compute and print the volume of the box. Then read in 3 floating numbers that represent the sides of a triangle. On a new page, print the numbers. Compute and print the area of the triangle using the following formulas:

\[
\text{AREA} = S \times (S - A) \times (S - B) \times (S - C) \\
S = (A + B + C) / 2
\]
EXERCISES

1. Define the terms, field, record, and file. Give an example of what kind of information would be held in each.

2. Show the values of the variables after the execution of all the GET LIST instructions.

```
GET LIST (A,B,C);
GET LIST (D,E,B);
GET LIST (C,B);
GET SKIP LIST (E,A,C);
```

Data: 1,2,3,4,5,6,7
8,9,10,11,12
13,14,15,16

3. What would be printed with the following PUT LIST instructions?

```
PUT LIST ('ABC',25,'23+2+4',23+2+4);
PUT LIST ('PRICE','AREA');
PUT SKIP LIST ('ALL DONE');
```

4. What would be printed after the execution of the following instructions?

```
GET LIST (VALUE,COUNT,A);
GET LIST (COUNT,VALUE);
PUT LIST (VALUE,COUNT,'NOTHING','A','A');
GET LIST (A,VALUE,COUNT);
PUT LIST ('VALUE IS:','VALUE');
PUT LIST (COUNT,'IS THE COUNT');
GET SKIP LIST (COUNT);
GET LIST (VALUE,A,COUNT);
PUT SKIP LIST (COUNT+A,VALUE,A);
```

Data: 10,15,85,100,200
0,65,78,2
6,79,54,9

PROBLEMS

1. Write a PL/1 program to print out an address label with your name, address, and telephone number. The address label should be similar to the following:

RUTH A. BAKER
1125 S. ELM
KIRKLAND, IL 60185
815-344-7986
2. Write a PL/1 program to perform the following algorithm. Do not forget to declare all variables.

START
1. Write a heading.
2. Read 3 numbers of precision (3,1).
3. Triple space and print the 3 numbers labeled.
4. Calculate the sum of the 3 numbers.
5. Double space and print the sum.
END

3. Redo Problem 3 in Chapter 3 using GET PUT and GET LIST. Use labels to identify all values printed.
CHAPTER 5

EXERCISES

1. Which of the following logical expressions are true given
\( W = 20, X = 5, Y = 2, Z = 0 \)?

A. \( W = 85 \) OR \( X = 5 \) OR \( Z = -9 \)

B. \((X + Y) = 7\) AND \(X < Z\) AND \(W \) NOT \( = Z\)

C. \( Z = W \) OR \( Z = 0 \) AND \( Z = 2 \) OR \( X = (W - 5)\)

D. \( 20 < 18 \) AND \( Y = Z \) OR \( X \) NOT \(< Y\) AND \(Z \times W = 0\)

2. Code the logical expressions from Exercise 1 into PL/1 code.
(Use the IF portion of the IF-THEN statement)

3. Write the PL/1 code to perform each of the following flowchart segments:

[Flowcharts 5.3.A, 5.3.B, 5.3.C]

PROBLEMS

1. Draw a flowchart and write a PL/1 program to read, print, and calculate the sum of 50 integers. Print the sum when finished.
(Hint: Use a DO WHILE loop)

2. A) Draw a flowchart and write a PL/1 program which will:
   1. Read an unknown amount of numbers (fixed decimal) terminated by a 0.
   2. Print the numbers (one per line).
   3. Print the smallest and largest values (labeled).
   4. Print an error message if there are not any numbers to be read.

B) Redo part A) using a header count record to determine how many numbers are to be read.

3. Draw a flowchart and write a PL/1 program to read and print a set of records, each containing the following:

   \[
   \begin{align*}
   \text{PART\_NUMBER} & \quad \text{FIXED DEC (2)} \\
   \text{PRICE} & \quad \text{FIXED DEC (5,2)} \\
   \text{PART\_QUANTITY} & \quad \text{FIXED DEC (3)}
   \end{align*}
   \]

Compute and print the average cost (labeled). Include column headers and a trailer message specifying end of file was reached. (Note: Check for the possibility of a no data error).
CHAPTER 6

EXERCISES

1. List the four goals of good programming.

2. What are three ways to increase the maintainability of a program?

3. One of the requirements of structured programming is the use of the three allowable structures. Draw an example flowchart for each of the structures; then list and define the other three requirements for structured programming.

PROBLEMS

1. Write the comments that belong at the head of the program you wrote for problem 3 in Chapter 5. Include the FUNCTION, INPUT, OUTPUT, AND NOTES.

2. A) Given the following PL/1 code and input data, what output would be produced by the PUT LIST and CHECK instructions?

/* PROBLEM 2 PROGRAM USING CHECK */
(CHECK (TOTAL_RECEIVED, MAX_DUE));
PROB2 : PROCEDURE OPTIONS (MAIN);
DECLARE ACCT_NUM FIXED DECIMAL (5),
AMOUNT-RECEIVED FIXED DECIMAL (5,2),
AMOUNT_DUE FIXED DECIMAL (5,2),
MAX_ACCT_NUM FIXED DECIMAL (5),
TOTAL_RECEIVED FIXED DECIMAL (6,2) INIT (0),
MAX_DUE FIXED DECIMAL (5,2) INIT (0),
EOF_FLAG BIT (1) INIT ('1'B);

ON ENDFILE (SYSIN) EOF_FLAG = 'O'B;
PUT LIST ('', '', 'ACCOUNT RECEIVABLE REPORT');
PUT SKIP (3) LIST ('', 'ACCOUNT NUMBER', 'RECEIVED', 'AMOUNT DUE');
GET LIST (ACCT_NUM,AMOUNT_RECEIVED,AMOUNT_DUE);
DO WHILE (EOF_FLAG = '1'B);
PUT SKIP LIST ('',ACCT_NUM,AMOUNT_RECEIVED,AMOUNT_DUE');
IF AMOUNT_DUE > MAX_DUE THEN
DO;
MAX_DUE = AMOUNT_DUE;
MAX_ACCT_NUM = ACCT_NUM;
END;
TOTAL_RECEIVED = TOTAL_RECEIVED + AMOUNT_RECEIVED;
GET LIST (ACCT_NUM,AMOUNT_RECEIVED,AMOUNT_DUE);
END;

PUT SKIP (3) LIST ('', '', 'TOTAL RECEIVED IS',TOTAL_RECEIVED);
PUT SKIP LIST ('', '', 'ACCOUNT NUMBER:', MAX_ACCT_NUM,
 'HAS THE LARGEST AMOUNT DUE:', MAX_DUE);
END;
DATA: 
28956, 150.00, 100.00
55555, 300.00, 100.00
67890, 900.00, 0.00
12345, 15.00, 250.00
44444, 650.00, 20.00
47321, 10.00, 900.00

B) Write the comments for the head of the program in part A).
CHAPTER 7

EXERCISES

1. Given the following flowcharts, what values would be printed?
   A. [Flowchart 7.1.A]
   B. [Flowchart 7.1.B]
   Record Format:  
   col 1-9 Social Security Number
   col 10-11 Age
   col 12 Sex 'F' or 'M'
   Data:  
   332681957,35,M
   884689357,50,F
   123456789,18,M
   987654321,16,M
   567891234,40,F
   567894321,32,M
   432156789,65,F
   432198765,03,M
   C. [Flowchart 7.1.C]
   Record Format:  
   col 1-4 Part number
   5-7 Quantity
   Data:  
   7872,53
   7872,2
   7872,100
   9999,1
   9999,16
   1000,83
   6767,50
   6767,25

2. List the three important rules for using nested IF's.

3. Define the following report logic terms:
   A. detail line
   B. control break
   C. footings

PROBLEMS

1. Write a PL/1 program for each of the flowcharts in Exercise 1.A) and 1.B).

2. Write the PL/1 code using SELECT statements to implement the following case structure.
   [Flowchart 7.2.P]
3. The Computer Science department needs a report that lists the students enrolled in each course. Start each course list on a new page, and include subtotals for the total number of students in the course, and the number of undergraduate and graduate students. At the end of the report, print the grand total of students taking Computer Science courses.

Record format  
1-9 Social Security Number  
10-12 Class Number  
13 Type of Student  
  G = Graduate  
  U = Undergraduate
START

\[ X \leftarrow 4 \]

\[ X \leftarrow (3 \times X) + 3 \]

WRITE

\[ X \]

STOP
Flowchart 1.1. B

START

I ← 0
S ← 0
N ← 5

I < N?

NO

YES

S ← I + S
I ← I + 1

WRITE S

STOP
Flowchart 1.1.6

START

READ A, B, C

YES C < A+B NO

X = A*B+C

X = (A+C)/B

WRITE X

STOP
Flowchart 5.3.A

- **YES** \( X < Y ? \) \( \rightarrow \)
  - \( C \leftarrow X + 3 \)
  - \( X \leftarrow X + 1 \)

- **NO** \( \rightarrow \)
  - \( C \leftarrow Y + 3 \)
Flowchart 5.3.B

1. VALUE ← 25
2. CNT ← 0
3. If CNT > 25 then NO, else YES
4. If YES then
   a. VALUE ← VALUE + 1
   b. CNT ← CNT + 1
5. End If
6. End If
Flowchart 5.3.C

\[
\begin{align*}
\text{POS_CNT} & \leftarrow 0 \\
\text{NEG_CNT} & \leftarrow 0 \\
\text{NUM_RD} & \leftarrow 0 \\
\text{READ} \\
\text{READ_CNT} \\
\text{READ} \\
\text{INTEGER} \\
\text{NUM_RD} & \leftarrow \text{READ_CNT} \\
\text{WRITE} \\
\text{INTEGER} \\
\text{INTEGER} & \leftarrow 0 \\
\text{POS_CNT} & \leftarrow \text{POS_CNT} + 1 \\
\text{NEG_CNT} & \leftarrow \text{NEG_CNT} + 1 \\
\text{NUM_RD} & \leftarrow \text{NUM_RD} + 1 \\
\text{READ} \\
\text{INTEGER} \\
\text{WRITE} \\
\text{NEG_CNT} \\
\text{WRITE} \\
\text{POS_CNT}
\end{align*}
\]
Flowchart 7.1.1

START

I ← 50
J ← 20
K ← 3

I > J?

I < K?

WRITE "MINIMUM" I

I > K?

WRITE "MAXIMUM" I

STOP
START
READ A RECORD

ENDFILE? YES

NO

MALE? YES

NO

AGE < 21 YES

GROUP1 ← GROUP1 + 1

NO

AGE < 35 YES

GROUP2 ← GROUP2 + 1

NO

AGE < 50 YES

GROUP3 ← GROUP3 + 1

NO

GROUP4 ← GROUP4 + 1

WRITE RECORD

READ A RECORD
WRITE "GROUP1"
GROUP1

WRITE "GROUP2"
GROUP2

WRITE "GROUP3"
GROUP3

WRITE "GROUP4"
GROUP4

STOP
Flowchart 7.1.C

START

READ A RECORD

ENDFILE

YES

SAVE PART_NUM

NO

NEW PART_NUM OR ENDFILE

YES

TOT_QUAN ← TOT_QUAN + QUANTITY

READ A RECORD

NO

WRITE PART_NUM TOT_QUAN

TOT_QUAN ← 0

STOP
Flowchart 7.2

READ
CLASS_CODE

CLASS_CODE=1
CLASS1 ←
CLASS1 + 1

CLASS_CODE=2
CLASS2 ←
CLASS2 + 1

CLASS_CODE=3
CLASS3 ←
CLASS3 + 1
CHAPTER 1 ANSWERS

EXERCISES

1. A) 15
   B) 0 1 3 6 10
   C) 100

2. A) START
   X <-- 4
   X <-- 3 * X + 3
   WRITE X
   END
   B) START
   I <-- 0
   S <-- 0
   N <-- 0
   DO WHILE (I < N)
      S <-- I + S
      I <-- I + 1
      WRITE S
   ENDDO
   END
   C) START
      READ A, B, C
      IF (C < A + B)
         X <-- A * B + C
      ELSE
         X <-- (A + C) / B
      ENDF
      WRITE X
   END

3. A) [Flowchart 1.3.A]
   B) [Flowchart 1.3.B]
   C) [Flowchart 1.3.C]

PROBLEMS

1. A) START
   N <-- 1
   SUM <-- 0
   DO WHILE (N < 100)
      READ X
      WRITE X
      SUM <-- SUM + X
      N <-- N + 1
   ENDDO
   WRITE SUM
   END
   [Flowchart 1.1.A.P]  
   B) START
      CTR <-- 0
      N <-- 0
      SUM <-- 0
      READ X
      DO WHILE (N < 10)
         IF (X > 0)
            SUM <-- SUM + X
            CTR <-- CTR + 1
         ENDF
         N <-- N + 1
         READ X
      ENDDO
      AVE <-- SUM / CTR
      WRITE AVE
   END
   [Flowchart 1.1.B.P]
C) START
    READ MAX
    CTR <-- 1
    DO WHILE (CTR < 5)
        READ X
        IF (X > MAX)
            MAX <-- X
        ENDIF
        CTR <-- CTR + 1
    ENDDO
    WRITE MAX
END

[Flowchart 1.1.C.P]
CHAPTER 2 ANSWERS

EXERCISES

1. A) 8 halfwords, 16 bytes, 128 bits
   B) The binary number system is used by the computer.
   C) Compilation - the compiler has the following three functions:
      - checks for errors and warnings in PL/1 code,
      - translates PL/1 code to machine code,
      - and generates a copy of the PL/1 code

2. A) invalid: too long
   B) valid
   C) invalid: too long
   D) invalid: starts with a number
   E) invalid: embedded blank
   F) valid

3. A) 7
   B) -6
   C) -3
   D) 48

4. A) WAGES= 200  TAX= 20  PAY= 180
   B) NUM1 = 4  NUM2= 40  NUM3= 42  AVE= 28
   C) HOURS= 24  MINUTES= 1440  SECONDS= 86400

PROBLEMS

1. SUM: PROCEDURE OPTIONS (MAIN);
   DECLARE A;
   DECLARE B;
   DECLARE TOTAL;
   A = 245;
   B = 73;
   TOTAL = A + B;
   PUT DATA;
   END SUM;
2. 
NUMSEM: PROCEDURE OPTIONS (MAIN);
   DECLARE CURRENT;
   DECLARE HOUR_SEM;
   DECLARE GRAD_REQ;
   DECLARE NUM_SEM;
   CURRENT = 20;
   HOUR_SEM = 15;
   GRAD_REQ = 125;
   NUM_SEM = (GRAD_REQ - CURRENT) / HOUR_SEM;
   PUT DATA;
   END NUMSEM;

3. 
GPA: PROCEDURE OPTIONS (MAIN);
   DECLARE AUDPTS;
   DECLARE CALCPTS;
   DECLARE ENGPTS;
   DECLARE TOTPTS;
   DECLARE TOTHRS;
   DECLARE GPA;
   AUDPTS = 4 * 4;
   CALCPTS = 2 * 3;
   ENGPTS = 3 * 2;
   TOTPTS = AUDPTS + CALCPTS + ENGPTS;
   TOTHRS = 4 + 3 + 2;
   GPA = TOTPTS / TOTHRS;
   PUT DATA;
   END GPA;
CHAPTER 3 ANSWERS

EXERCISES

1. **ATTRIBUTE**
   - scale  
   - base  
   - precision  
   - initial

   **OPTIONS**
   - fixed, float  
   - decimal, binary

2. A) DECLARE SUM FIXED DECIMAL (6) INITIAL (0);
   B) DECLARE AVERAGE FIXED DECIMAL (4,2);
   C) DECLARE INTEREST FLOAT DECIMAL (8) INITIAL (62.45);
   D) DECLARE COUNTER FIXED BINARY (15) INITIAL (1);

3. A) 51
   B) -14
   C) 81
   D) 1
   E) -8

4. A) NUMBER1= 345.0000 NUMBER2= 123
   B) NUMBER1= .0246 NUMBER2= 56789
   C) NUMBER1= 3456.0000 NUMBER2= 2
   D) NUMBER1= 123.4560 NUMBER2= 12340
   E) NUMBER1= 2345.6000 NUMBER2= 0
   F) NUMBER1= .1230 NUMBER2= 9876

PROBLEMS

1. **MAXNUM**: PROCEDURE OPTIONS (MAIN);
   DECLARE PRICE1 FIXED DECIMAL (6,2),
   DECLARE PRICE2 FIXED DECIMAL (6,2),
   DECLARE PRICE3 FIXED DECIMAL (6,2),
   DECLARE MAXPRICE FIXED DECIMAL (6,2);

   GET LIST (PRICE1);
   GET LIST (PRICE2);
GET LIST (PRICE3);

MAXPRICE = MAX (PRICE1, PRICE2, PRICE3);

PUT DATA (MAXPRICE);
<
END MAXNUM;

2.

MEASURE: PROCEDURE OPTIONS (MAIN);
DECLARE BOXSIDE1 FIXED,
  BOXSIDE2 FIXED,
  BOXSIDE3 FIXED,

  TRISIDE1 FLOAT,
  TRISIDE2 FLOAT,
  TRISIDE3 FLOAT,
  S   FLOAT,
  AREA  FLOAT;

/* READ AND PRINT THE SIDES OF THE BOX*/
GET LIST (BOXSIDE1);
GET LIST (BOXSIDE2);
GET LIST (BOXSIDE3);

PUT DATA (BOXSIDE1, BOXSIDE2, BOXSIDE3);

/* COMPUTE AND PRINT THE VOLUME OF THE BOX */
VOLUME = BOXSIDE1 * BOXSIDE2 * BOXSIDE3;
PUT DATA (VOLUME);

/* READ AND PRINT THE SIDES OF THE TRIANGLE */
GET LIST (TRISIDE1);
GET LIST (TRISIDE2);
GET LIST (TRISIDE3);

PUT PAGE DATA (TRISIDE1, TRISIDE2, TRISIDE3);

/* COMPUTE AND PRINT THE AREA OF THE TRIANGLE */
S = (TRISIDE1 + TRISIDE2 + TRISIDE3) * .5
AREA = SQR(S*(S-TRISIDE1)*(S-TRISIDE2)*(S-TRISIDE3));

PUT DATA (AREA);
END MEASURE;
CHAPTER 4 ANSWERS

EXERCISES

1. **field** - piece of information contained in a record, i.e. customer address
   **record** - one or more fields grouped together, i.e. employee info. record
   **file** - a group of records each containing the same fields, i.e. inventory record file

2. A) 14
   B) 8
   C) 15
   D) 4
   E) 13

3.

   | ABC | __________ | 25 | __________ | 23+2+4 | __________ | 29 | __________ | PRICE | __________ |
   ---|-------|--------|----|--------|--------|----|--------|-------|-----------|
   | AREA | __________ | __________ | __________ | __________ | __________ | __________ | __________ | __________ | __________ |
   | ALL DONE | __________ | __________ | __________ | __________ | __________ | __________ | __________ | __________ | __________ |

4.

   | 200 | __________ | 100 | __________ | NOTHING | __________ | A | __________ | 85 | __________ |
   ---|-----|--------|----|--------|----------|--------|---|--------|----|-----------|
   | 85 | __________ | 200 | __________ | 100 | __________ | VALUE IS: | __________ | 78 | __________ |
   | 65 | __________ | IS THE COUNT | __________ | __________ | __________ | __________ | __________ | __________ | __________ |
   | 63 | __________ | 79 | __________ | 54 | __________ | __________ | __________ | __________ | __________ |

PROBLEMS

1. /* PROBLEM1 - PRINT ADDRESS LABEL */
   **PROB1: PROCEDURE OPTIONS (MAIN);**
   PUT LIST (‘RUTH A. BAKER’); PUT SKIP LIST (‘1125 S. ELM’); PUT SKIP LIST (‘KIRKLAND, IL. 60185’); PUT SKIP (2) LIST (‘815-344-7986’);
   END PROB1;

2. /* PROBLEM2 - PROCESS 3 NUMBERS */
   **PROB2: PROCEDURE OPTIONS (MAIN);**
   DECLARE A FIXED DECIMAL (3,1), B FIXED DECIMAL (3,1), C FIXED DECIMAL (3,1), SUM FIXED DECIMAL (5,1);
   GET LIST (A,B,C);
   PUT LIST (' ', 'REPORT FOR PROBLEM2');
PUT SKIP (3) LIST (A,B,C);
SUM = A + B + C;
PUT SKIP (2) LIST (SUM);
END PROB2;

3.

/* MEASURE: PROCEDURE OPTIONS (MAIN);
DECLARE BOXSIDE1 FIXED,
BOXSIDE2 FIXED,
BOXSIDE3 FIXED,
VOLUME FIXED,
TRISIDE1 FLOAT,
TRISIDE2 FLOAT,
TRISIDE3 FLOAT,
S FLOAT,
AREA FLOAT;

/* READ AND PRINT THE SIDES OF THE BOX */
GET LIST (BOXSIDE1, BOXSIDE2, BOXSIDE3);
PUT LIST ('DIMENSIONS OF BOX', BOXSIDE1, BOXSIDE2, BOXSIDE3);

/* COMPUTE AND PRINT VOLUME OF BOX */
VOLUME = BOXSIDE1 * BOXSIDE2 * BOXSIDE3;
PUT SKIP LIST ('THE VOLUME OF THE BOX IS', VOLUME);

/* READ AND PRINT THE SIDES OF THE TRIANGLE */
GET LIST (TRISIDE1, TRISIDE2, TRISIDE3);
PUT PAGE LIST ('SIDES OF TRIANGLE', TRISIDE1, TRISIDE2, TRISIDE3);

/* COMPUTE AND PRINT THE AREA OF THE TRIANGLE */
S = (TRISIDE1 + TRISIDE2 + TRISIDE3) * .5;
AREA = SQRT(S*(S-TRISIDE1)*(S-TRISIDE2)*(S-TRISIDE3));
PUT SKIP LIST ('AREA OF TRIANGLE', AREA);
END MEASURE;
CHAPTER 5 ANSWERS

EXERCISES

1. A) True
   B) False
   C) True
   D) False

2. A) IF \((W = 85 \land X = 5 \land Z = -9)\) THEN
   B) IF \(((X + Y) = 7 \land X < Z \land W \leq Z)\) THEN
   C) IF \((Z = W \land Z = 0 \land Z = 2 \land X = (W - 5))\) THEN
   D) IF \((20 < 18 \land Y = Z \land X < (Y \& Z \land w) = 0)\) THEN

3. A) IF \(X < Y\) THEN
    
    DO;
    C = X + 3;
    X = X + 1;
    END;
    ELSE
    C = Y + 3;
    END;

    B) VALUE = 25;
    CNT = 0;
    DO WHILE (CNT <= 25);
    VALUE = VALUE - 1;
    CNT = CNT + 1;
    END;

    C) POS_CNT = 0;
    NEG_CNT = 0;
    NUM_READ = 0;
    GET LIST READ_CNT;
    GET LIST (INTEGER);
    DO WHILE(NUM_READ <= READ_CNT);
    PUT SKIP LIST (INTEGER);
    IF (INTEGER >= 0) THEN
    POS_CNT = POS_CNT + 1;
    ELSE
    NEG_CNT = NEG_CNT + 1;
    NUM_READ = NUM_READ + 1;
    GET LIST (INTEGER);
    END;

    PUT SKIP LIST ("NUMBER OF POSITIVE INTEGERS:",POS_CNT);
    PUT SKIP LIST ("NUMBER OF NEGATIVE INTEGERS:",NEG_CNT);

PROBLEMS

1. [Flowchart 5.1.P]

   /* PROBLEM1 - READ, PRINT & SUM 50 INTEGERS */
   PROB1: PROCEDURE OPTIONS (MAIN);
   DECLARE SUM FIXED DECIMAL (6) INIT (0),
I FIXED DECIMAL (2) INIT (0),
NUMB_READ FIXED DECIMAL;
DO WHILE (I ^= 50);
    GET LIST (NUMB_READ);
    PUT SKIP LIST (NUMB_READ);
    SUM = SUM + NUMB_READ;
    I = I + 1;
END;
    PUT SKIP LIST (SUM);
END PROB1;

2. A) [Flowchart 5.2.A.P]

/* PROBLEM 2A - FIND THE SMALLEST AND LARGEST */
PROB2A: PROCEDURE OPTIONS (MAIN);
DECLARE NUMB FIXED DECIMAL, 
MAX FIXED DECIMAL, 
MIN FIXED DECIMAL, 
READ_FLAG FIXED BINARY (1) INIT (0);

MAX = -99999;
MIN = 99999;
GET LIST (NUMB);

DO WHILE (NUMB ^= 0);
    READ_FLAG = 1;
    PUT SKIP LIST (NUMB);
    IF (NUMB > MAX) THEN
        MAX = NUMB;
    IF (NUMB < MIN) THEN
        MIN = NUMB;
    GET LIST (NUMB);
END;
IF (READ_FLAG = 0) THEN
    PUT SKIP LIST ('ERROR - NO DATA READ');
ELSE
    DO;
        PUT SKIP LIST ('THE SMALLEST NUMBER IS:',MIN);
        PUT SKIP LIST ('THE LARGEST NUMBER IS:',MAX);
    END;
END PROB2A;

B) [Flowchart 5.2.B.P]

/* PROBLEM 2B - USING A HEADER COUNT */
PROB2B: PROCEDURE OPTIONS (MAIN);
DECLARE NUMB FIXED DECIMAL, 
MAX FIXED DECIMAL INIT (-99999), 
MIN FIXED DECIMAL INIT (99999), 
COUNT FIXED DECIMAL INIT (0), 
HDR_CNT FIXED DECIMAL;
GET LIST (HDR_CNT);
DO WHILE (HDR_CNT <= COUNT);
    GET LIST (NUMB);
    PUT SKIP LIST (NUMB);
    IF (NUMB > MAX) THEN
        MAX = NUMB;
    IF (NUMB < MAX) THEN
        MIN = NUMB;
    COUNT = COUNT + 1;
END;

IF (HDR_CNT = 0) THEN
    PUT SKIP LIST ('ERROR - NO DATA READ');
ELSE
    DO;
        PUT SKIP LIST ('THE LARGEST NUMBER IS:', MAX);
        PUT SKIP LIST ('THE SMALLEST NUMBER IS:', MIN);
    END;
END;

C). [Flowchart 5.3.P]

/* PROBLEM 3 AVERAGE COST */
PROB3: PROCEDURE OPTIONS (MAIN);
    DECLARE DATA_FLAG FIXED BINARY (1) INIT (0),
             EOF_FLAG FIXED BINARY (1) INIT (0),
             PART_NUM FIXED DECIMAL (2),
             PRICE FIXED DECIMAL (5,2),
             QUAN FIXED DECIMAL (3),
             COST FIXED DECIMAL (8,2),
             TOT_COST FIXED DECIMAL (10,2) INIT (0),
             TOT_QUAN FIXED DECIMAL (5) INIT (0),
             AVE FIXED DECIMAL (5,2);
ON ENDFILE (SYSIN) EOF_FLAG = 1;
PUT PAGE LIST (' ', 'PART NUMBER', 'PRICE', 'QUANTITY');
GET LIST (PART_NUM,PRICE,QUAN);
DO WHILE (EOF_FLAG = 0);
    PUT SKIP LIST (' ', PART_NUM,PRICE,QUAN);
    COST = QUAN * PRICE;
    TOT_COST = TOT_COST + COST;
    TOT_QUAN = TOT_QUAN + QUAN;
    DATA_FLAG = 1;
    GET LIST (PART_NUM,PRICE,QUAN);
END;

IF (DATA_FLAG = 1) THEN
    DO;
        AVE = TOT_COST / TOT_QUAN;
        PUT SKIP (2) LIST ('THE AVERAGE COST IS:', AVE);
        PUT SKIP LIST ('END OF FILE WAS REACHED');
    END;
ELSE

PUT SKIP LIST ('ERROR - NO DATA READ');
END PROB3;
CHAPTER 6 ANSWERS

EXERCISES

1. **accuracy, reliability, readability, maintainability**

2. **A)** avoid using literals (hardcoding)
   **B)** use meaningful names for all variables
   **C)** indentation should be used to show the subordinate lines of code dependent upon conditions

3. [Flowchart 6.3]
   **TOP DOWN DEVELOPMENT** - the development of a program by starting with a general problem and moving toward more specific details.
   **TOP-DRIVEN LOOPS** - the condition of a loop is checked at the top of the loop.
   **ONE ENTRANCE/ONE EXIT** - any program or any loops within a program can only be entered in one place and exited in one place.

PROBLEMS

1.

```plaintext
/***************************************************************************/
/* */
/* FUNCTION: THIS PROGRAM PRODUCES A REPORT LISTING INFORMATION */
/* ABOUT PART ORDERS. */
/* */
/* */
/* INPUT: FOR EACH PART ORDERED A RECORD WITH: */
/* */
/* 1. PART NUMBER */
/* */
/* 2. COST */
/* */
/* 3. PART QUANTITY */
/* */
/* OUTPUT: THE INPUT WITH IDENTIFYING COLUMN HEADERS, AND THE */
/* AVERAGE COST. A MESSAGE IS ALSO PRINTED WHEN */
/* END-OF-FILE IS REACHED. */
/* */
/* NOTES: AVERAGE COST = TOTAL COST / TOTAL QUANTITY */
/***************************************************************************/
```

2. **A)**

**ACCOUNTS RECEIVABLE REPORT**

<table>
<thead>
<tr>
<th>ACCOUNT NUMBER</th>
<th>RECEIVED</th>
<th>AMOUNT DUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>28956</td>
<td>150.00</td>
<td>100.00</td>
</tr>
<tr>
<td>CHECK</td>
<td>MAX_DUE</td>
<td>TOTAL_RECEIVED</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>----------------</td>
</tr>
<tr>
<td></td>
<td>100.00</td>
<td>150.00</td>
</tr>
<tr>
<td>55555</td>
<td>300.00</td>
<td>100.00</td>
</tr>
<tr>
<td>67890</td>
<td>900.00</td>
<td>.00</td>
</tr>
<tr>
<td>12345</td>
<td>15.00</td>
<td>250.00</td>
</tr>
<tr>
<td>44444</td>
<td>650.00</td>
<td>20.00</td>
</tr>
<tr>
<td>47321</td>
<td>10.00</td>
<td>900.00</td>
</tr>
<tr>
<td></td>
<td>900.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2015.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2025.00</td>
<td></td>
</tr>
</tbody>
</table>

TOTAL RECEIVED IS: 2025.00
ACCOUNT NUMBER: 47321
HAS THE LARGEST AMOUNT DUE: 900.00

2. B)

/***************************************************************/
/* *FUNCTION: THIS PROGRAM PRODUCES A REPORT ON THE PAYMENTS OF */
/* AN ACCOUNTS RECEIVABLE SYSTEM.*/
/* */
/* INPUT: FOR EACH PAYMENT A RECORD CONTAINING THE ACCOUNT */
/* NUMBER, AMOUNT OF PAYMENT, AND AMOUNT STILL DUE ON */
/* THE ACCOUNT.*/
/* */
/* OUTPUT: THE INPUT, TOTAL AMOUNT RECEIVED, AND ACCOUNT */
/* NUMBER WITH LARGEST AMOUNT STILL DUE.*/
/* */
/* NOTES: NONE*/
/***************************************************************/
CHAPTER 7 ANSWERS

EXERCISES

1. A) MAXIMUM 50

   B) 332681957 35 2
      123456789 18 2
      987654321 16 2
      567894321 32 2
      432198765 3 2

      GROUP1 3
      GROUP2 1
      GROUP3 1
      GROUP4 0

C) 7872 155
   9999 17
   1000 33
   6767 75

2. 1. When a false condition is encountered in the IF-THEN, no more tests are made; control passes to the corresponding ELSE.

2. Each ELSE corresponds to the most immediately preceding IF-THEN that does not already have a corresponding ELSE.

3. Each IF-THEN must have a corresponding ELSE when the next higher level IF-THEN has a corresponding ELSE.

3. detail line - a single line of output, such as client information, or an employee's wages.

control block - point at which one group of records ends and another begins.

footings - subtotals for each group of records

PROBLEMS

1. A)

/* PROBLEM 1A NESTED IF'S */

PROB1A: PROCEDURE OPTIONS (MAIN);

   DECLARE I FIXED DECIMAL,
        J FIXED DECIMAL,
        K FIXED DECIMAL;

   I = 50;
   J = 20;
   K = 3;

   IF (I > J) THEN
DO;
  IF (I > K) THEN
    PUT SKIP LIST ('MAXIMUM',I);
  END;
ELSE
  DO;
    IF (I < K) THEN
      PUT SKIP LIST ('MINIMUM',I);
    END;
  END PROB1A;

B)
/* PROBLEM 1B NESTED IF'S */
PROB1B: PROCEDURE OPTIONS (MAIN);
DECLARE SSN FIXED DECIMAL (9),
    AGE FIXED DECIMAL (2),
    SEX FIXED DECIMAL (1),
    EOF_FLAG FIXED BINARY (1) INIT (0),
    GROUP1 FIXED DECIMAL (3) INIT (0),
    GROUP2 FIXED DECIMAL (3) INIT (0),
    GROUP3 FIXED DECIMAL (3) INIT (0),
    GROUP4 FIXED DECIMAL (3) INIT (0);
ON ENDFILE (SYSIN) EOF_FLAG = 1;
GET LIST (SSN,AGE,SEX);
DO WHILE (EOF_FLAG = 0);
  IF (SEX = 2) THEN
    DO;
      IF (AGE < 21) THEN
        GROUP1 = GROUP1 + 1;
      ELSE
        IF (AGE < 35) THEN
          GROUP2 = GROUP2 + 1;
        ELSE
          IF (AGE < 50) THEN
            GROUP3 = GROUP3 + 1;
          ELSE
            GROUP4 = GROUP4 + 1;
          END;
        END;
      END;
    END;
  END;
GET LIST (SSN,AGE,SEX);
END;
PUT SKIP LIST ('AGE GROUP1',GROUP1);
PUT SKIP LIST ('AGE GROUP2',GROUP2);
PUT SKIP LIST ('AGE GROUP3',GROUP3);
PUT SKIP LIST ('AGE GROUP4',GROUP4);
END PROB1B;

2.
GET LIST (CLASS_CODE);
SELECT;
  WHEN (CLASS_CODE = 1)
    DO;
      CLASS1 = CLASS1 + 1;
    END;
WHEN (CLASS_CODE = 2)
    DO;
    CLASS2 = CLASS2 + 1;
    END;
WHEN (CLASS_CODE = 3)
    DO;
    CLASS3 = CLASS3 + 1;
    END;
END;

3. /* PROBLEM 3 COURSE LIST */
PROB3: PROCEDURE OPTIONS (MAIN);
DECLARE SSN FIXED DECIMAL (9),
        CLASS_NUM FIXED DECIMAL (3),
        STUD_TYPE FIXED DECIMAL (1),
        CRSE_TOTAL FIXED DECIMAL (3) INIT (0),
        TOT_GRAD FIXED DECIMAL (3) INIT (0),
        TOT_UNGRAD FIXED DECIMAL (3) INIT (0),
        EOF_FLAG FIXED BINARY (1) INIT (0),
        SAVE_CLASS FIXED DECIMAL (3);
ON ENDFILE (SYSIN) EOF_FLAG = 1;
GET LIST (SSN,CLASS_NUM,STUD_TYPE);
DO WHILE (EOF_FLAG = 0);
    PUT PAGE LIST (' ', ' ', 'STUDENT COURSE REPORT');
    PUT SKIP LIST (' ', ' ', 'SSN', 'CLASS', 'STUDENT NUMBER');
    PUT SKIP LIST (' ');
    SAVE_CLASS = CLASS_NUM;
    DO WHILE (SAVE_CLASS = CLASS_NUM & EOF_FLAG = 0);
        PUT SKIP LIST (' ', ' ', 'SSN', 'CLASS_NUM', 'STUD_TYPE');
        CRSE_TOTAL = CRSE_TOTAL + 1;
        IF (STUD_TYPE = 1) THEN
            TOT_GRAD = TOT_GRAD + 1;
        ELSE
            TOT_UNGRAD = TOT_UNGRAD + 1;
        GET LIST (SSN,CLASS_NUM,STUD_TYPE);
    END;
    PUT SKIP LIST ('CLASS TOTAL: ', CRSE_TOTAL);
    PUT SKIP LIST ('TOTAL NUMBER OF UNDERGRADUATE STUDENTS',
                   TOT_UNGRAD);
    PUT SKIP LIST ('TOTAL NUMBER OF GRADUATE STUDENTS', TOT_GRAD);
    GRAND_TOTAL = 0;
    CRSE_TOTAL = 0;
    TOT_GRAD = 0;
    TOT_UNGRAD = 0;
END;
PUT SKIP LIST ('CLASS GRAND TOTAL', GRAND_TOTAL);
END PROB3;
Flowchart 1.3.A

START

READ X, Y

WRITE X, Y

X < Y?

YES Z ← Y - X

NO Z ← X - Y

WRITE Z

STOP
Flowchart 1.3.B

START

READ N

CTR ← 0
PCTR ← 0

CTR < N? NO

YES

READ X

X < 0? YES

NO

PCTR ← PCTR + 1

CTR ← CTR + 1

WRITE CTR, PTR

STOP
Flowchart 1.3.C

START

READ
A, B

PRINT
A, B

C ← 2*B+A-3

WRITE
C

STOP
Flowchart 1.1.A.P

START

N ← 1
SUM ← 0

N < 100?

READ X

WRITE X

SUM ← SUM + X

N ← N + 1

WRITE SUM

STOP
Flowchart 1.1.B.P

START

CTR ← 0
N ← 0
SUM ← 0

READ X

N < 10?

YES

X > 0

YES

SUM ← SUM + X
CTR ← CTR + 1

NO

NO

N ← N + 1

READ X

AVE ← SUM/CTR

WRITE AVE

STOP
START

READ MAX

CTR ← 1

CTR < 5?

NO

YES

READ X

X > MAX?

NO

YES

MAC ← X

CTR ← CTR + 1

WRITE MAX

STOP
Flowchart 5.1.P

START

SUM ← 0
I ← 0

I = 50?

YES

NO

READ
INTEGER

WRITE
INTEGER

SUM ← SUM + INTEGER

I ← I + 1

WRITE
SUM

STOP
Flowchart 5.2.A.P

START

READ NUMBER

READ_FLAG ← NO

MAX ← 9999
MIN ← -9999

NUMBER = 0?

READ_FLAG ← YES

WRITE NUMBER

NUMBER > MAX?

MAX ← NUMBER

NUMBER < MIN?

MIN ← NUMBER

READ NUMBER
Flowchart 5.2.A.P Continued

YES  READ_FLAG  NO

WRITE ERROR MESSAGE

WRITE MAX, MIN

STOP
START

COUNT ← 0
MAX ← 9999
MIN ← -9999

READ HDR_CNT

HDTR_CNT = COUNT?

READ NUMB

WRITE NUMB

NUMB > MAX?

MAX ← NUMB

NUMB < MIN?

MIN ← NUMBER

COUNT ← COUNT + 1
Flowchart 5.2.B.P Continued

- YES
  - HDR_CNT = 0
    - WRITE ERROR MESSAGE
    - NO
- WRITE MAX, MIN
- STOP
START
DATA_FLAG ← NO

TOT_COST ← 0
TOT_QUAN ← 0

WRITE COLUMN HEADERS

READ PART_NUM, PRICE, QUAN

EOF?

YES

WRITE RECORD

COST ← QUAN * PRICE

TOT_COST ← TOT_COST + COST

TOT_QUAN ← TOT_QUAN + QUAN

DATA_FLAG ← YES

READ PART_NUM, PRICE, QUAN

NO
Flowchart 5.3.P Continued

IF DATA_FLAG = YES

WRITE ERROR MESSAGE

ELSE

AVE ← TOT_COST / TOT_QUAN

WRITE AVE

WRITE TRAILER MESSAGE

STOP
Flowchart 6.3

PAY ← WAGES + TIPS

NET_PAY ← PAY - TAXES

WRITE NET_PAY

YES

N > 50?

NO

Z ← N - 25

Z ← N + 25

COUNT ← 0

INDEX ← 10

INDEX = 0

YES

NO

COUNT ← COUNT + INDEX

INDEX ← INDEX - 1
CHAPTER 1 TEST QUESTIONS

SHORT ANSWER

1. What are the three parts of a problem that should be considered?

2. What is an algorithm?

3. List two methods of expressing algorithms.

MULTIPLE CHOICE

1. Which of the following structures is not allowable in a structured flowchart?

   (a) IF-THEN.
   (b) DO-WHILE.
   (c) DO UNTIL.
   (d) IF-THEN-ELSE.

2. Which one of the following actions would be represented in a process flowchart box?

   (a) print the page number of a report.
   (b) add gross pay to year to date pay.
   (c) is the customer bill paid?
   (d) read a student record.

3. What kind of geometric symbol represents a decision in flowcharting?

   (a) diamond.
   (b) circle.
   (c) square.
   (d) rectangle.

PLAYING COMPUTER

What values are printed in each of the algorithms presented in the following pseudocode?

A) START
   X <-- 8
   Y <-- 32
   IF (Y < X)
     Z <-- Y - X
   ELSE
     Z <-- X - Y
   ENDIF
   WRITE X, Y, Z
   END

B) START
   NEG <-- 0
   POS <-- 0
   I <-- 0
   READ COUNT
   DO WHILE (COUNT NOT = I)
     READ VALUE
     IF (VALUE < 0)
       NEG <-- NEG + 1
     ELSE
       POS <-- POS + 1
     ENDIF
     I <-- I + 1
   END
Input Data: 10, -1, 300, 8, -50
0, -13, 13, 2, 99, 225

SHORT PROBLEMS

Draw a flowchart and write pseudocode for the following problems.

1. You are given a list of positive integers. Read and print each integer, and then print the largest and smallest integer in the list. Assume that the first integer read is the number of integers in the list.

2. Read in the length of each side of triangle A, and then the lengths for triangle B. Calculate the volume for each triangle. If the volume for triangle A is larger than 100 print the volume. If triangle B has a volume equal to the volume of triangle A, print volume of triangle B.
CHAPTER 2 TEST QUESTIONS

SHORT ANSWER

1. Define a bistable device.

2. List the three major functions of the compiler.

3. What two statements must any program always contain?

4. List the three types of characters that can compose a label.

MULTIPLE CHOICE

1. Which of the following binary numbers is equivalent to 9 in decimal?
   (a) 1000
   (b) 1001
   (c) 0111
   (d) 0101

2. How many bytes are in three doublewords?
   (a) 3
   (b) 36
   (c) 24
   (d) 12

3. Which one of the following procedure statements is correct?
   (a) NUMBER3: PROCEDURE OPTIONS (MAIN);
   (b) NUMBER 345: PROCEDURE OPTIONS (MAIN);
   (c) NUMBER3: PROCEDURE OPTIONS (MAIN)
   (d) NUMBER PROCEDURE OPTIONS (MAIN)

4. Choose the invalid label.
   (a) CONSTANT_1
   (b) VALUE_A_B
   (c) INDEX
   (d) #1_PROB

PLAYING COMPUTER

Show what would be outputed after execution of the following programs.

A) /* THE PROGRAM COMPUTES THE PRICE OF A CAR */
   CAR_PRIC: PROCEDURE OPTIONS (MAIN);
   DECLARE BASE_PRICE;
   DECLARE AIR_COND;
   DECLARE RUST_PROOF;
DECLARE SNOW_TIRE;
DECLARE TOTAL;

BASE_PRICE = 7000;
AIR_COND = 300;
RUST_PROOF = 1500;
SNOW_TIRE = 100;

TOTAL = BASE_PRICE + AIR_COND + RUST_PROOF + (SNOW_TIRE * 4);

PUT DATA;
END CAR_PRIC;

B) /* THIS PROGRAM COMPUTES A PRODUCT */
PRODUCT: PROCEDURE OPTIONS (MAIN);

DECLARE VALUE1;
DECLARE VALUE2;
DECLARE VALUE3;
DECLARE PRODUCT;

VALUE1 = 3;
VALUE2 = VALUE1 * 8;
VALUE3 = VALUE2 * VALUE1;
PRODUCT = VALUE3 * VALUE1;

PUT DATA;
END PRODUCT;

SHORT PROBLEMS

1. Write a PL/1 program to compute and print the number of minutes in a week.

2. Code a PL/1 program to figure and print a payroll check amount. Pay = $250, Tips = $30, tax rate = 5%.

3. If you are driving your car at 50 miles per hour. Write a PL/1 program to compute how long it would take to travel 200 miles assuming you stopped for lunch for one hour.
CHAPTER 3 TEST QUESTIONS

SHORT ANSWER

1. Give an example of a BUILTIN function and show how it is used.

2. Explain the difference between explicitly and implicitly declared variables.

3. List 4 of the 5 attributes for numeric variables.

MULTIPLE CHOICE

1. What precision does "FIXED DECIMAL" default to?
   (a) (5)
   (b) (6,0)
   (c) (6,2)
   (d) (5,0)

2. If the variable NUMB1 is not declared, what are its default attributes?
   (a) FIXED DECIMAL (5)
   (b) FIXED BINARY (15)
   (c) FLOAT (15)
   (d) FLOAT DECIMAL (6).

3. Given A = 5, B = 3, C = 0, D = 10, evaluate the following expression: 3 + D * A ** C + D ** 2 / A + C - B
   (a) 63
   (b) 103
   (c) 9.66
   (d) 0

4. Given that X = 109.327 DECIMAL (7,4), what will the value of CONSTANT be after execution of CONSTANT = ROUND (X,2);
   (a) CONSTANT = 110
   (b) CONSTANT = 109.3300
   (c) CONSTANT = 109.3270
   (d) CONSTANT = 109.33

PLAYING COMPUTER

Given the following DECLARE statements, what would the PUT DATA; instruction print after each set of instructions?

DECLARE
   INDEX FIXED DECIMAL (4,2) INIT (0),
   A FIXED,
   B FIXED DECIMAL (5,2) INIT (100),
   ANSWER FIXED DECIMAL (6,2) INIT (0);
A)  
\[ A = B \times \text{INDEX}; \]
\[ B = A + B \times 100 \div 5 \times 5 - (\text{INDEX} + A); \]
\[ \text{INDEX} = \text{MAX}(A, B, \text{ANSWER}); \]
\[ \text{ANSWER} = \text{SQRT}(\text{INDEX}); \]
\[ \text{PUT DATA;} \]

B)  
\[ \text{PUT DATA;} \]
\[ \text{INDEX} = \text{INDEX} + 1.55; \]
\[ A = B - \text{INDEX}; \]
\[ \text{ANSWER} = \text{FLOOR}(A); \]
\[ \text{PUT DATA}(A, \text{ANSWER}); \]

C)  
\[ A = \text{MIN}(\text{INDEX}, B, \text{ANSWER}) + B; \]
\[ B = \text{SQRT}(B); \]
\[ \text{ANSWER} = B \div A + 10; \]
\[ \text{INDEX} = \text{TRUNC}(\text{ANSWER} \times 20 \times 1 \times .33); \]
\[ \text{PUT DATA}(B, \text{ANSWER}, \text{INDEX}); \]

SHORT PROBLEMS

1. Write the PL/1 statement(s) to set aside storage for two data fields, one a three digit integer to contain the number of tickets sold and the other to contain the cost per ticket (maximum value $999.99).

2. Write a PL/1 statement to calculate a value for BILLING_AMT that is the same as the value obtained from the following:

   Subtract 2,000 from units used, then multiply by .18, then add the value from UNITS_USED minus 500, then divide by 3.

3. Write the PL/1 statements to read a data line containing STUDENT_NUM, CLASS_LOAD, GPA, and then print these values at the top of a new page.

LONGER PROBLEMS

1. Write a PL/1 program to read in 5 different estimates for fixing the transmission on a car. Calculate the average estimate, and print it along with the highest and lowest bid. Print each value on a separate line with double spacing.

2. Write a PL/1 program to compute travel expenses for a business trip. Read in GAS_EXPENSE (max of 99.99), HOTEL_EXPENSE (max of 999.99), and TOLL_EXPENSE (max of 9.99). Compute the total travel expense for 3 salesmen. Also compute and print each total expense and compute and print the grand total to the nearest dollar.
CHAPTER 4 TEST QUESTIONS

1. Define the following terms.
   (a) input
   (b) output
   (c) field
   (d) record
   (e) file

2. What is the difference between LIST I/O and DATA I/O?

MULTIPLE CHOICE

1. When using a PUT DATA statement, output starts in columns:
   (a) 1,5,10,15,20
   (b) 1,24,48,71,96
   (c) 1,25,49,71,96
   (d) 0,10,20,30,40

2. GET SKIP LIST (VALUE1,VALUE2,VALUE3) does the following:
   (a) Skips to the next input record and begins reading.
   (b) Reads 3 fields and then skips to the next input record.
   (c) Reads 1 field, skips to the next record, reads another field, skips to the next record, and then reads another field.

3. Which of the following PUT LIST statements would cause an error?
   (a) PUT PAGE LIST (' ');
   (b) PUT LIST (VALUE1,NUM3,'STUDENT NUMBER');
   (c) PUT SKIP (2) LIST (A,D,44,ID NUMBER);
   (d) PUT LIST (VALUE2+3,'IS THE VALUE',' ');

PLAYING COMPUTER

1. Show the values of the variables after execution of all the GET LIST instructions:

   GET LIST (VAL1,VAL2,VAL3);
   GET LIST (VAL3);
   GET SKIP LIST (VAL4,VAL2,VAL1,VAL3);
   GET SKIP LIST (VAL3);

   Data:
   83
   210,88,97,6,84
   23,202,16,25,66,71
   73,81,62,103
2. What would be printed after each of the following instructions?

PUT LIST ('START', 834, '696.6', '');
PUT SKIP (2) LIST ('NEW LINE?', 'VAL1', 'VAL2', 99, 29+7+6, 'DONE?');
PUT LIST ('NO', 'NOT', 'YET', 385/5);
PUT SKIP LIST ('END OF PROBLEM 2');

3. What would be printed after the execution of the following instructions?

GET LIST (COUNT, INTEGER, VALUE);
GET LIST (INTEGER, INDEX);
PUT PAGE LIST ('HERE IS THE COUNT', COUNT, '', 'INDEX', INDEX);
GET SKIP LIST (INTEGER, 43+7, '43+7');
GET SKIP LIST (VALUE, INDEX, VALUE);
PUT LIST (VALUE+2, INDEX-3, 'THE END');

Data:
3, 9, 265, 1000
28, 68, 29, 33
205, 607, 22

SHORT PROBLEMS

1. Write a PL/1 program to perform the following:

A) Print a header message, such as PROBLEM2 - TEST#2
B) Read 2 integers (fixed constants) into memory locations. These integers represent the length and width of a rectangle respectively.
C) Print these numbers suitably labeled.
D) Compute and print the AREA of the rectangle.
E) Compute and print the PERIMETER of the rectangle.
F) Triple space and print a trailer message.

2. Write a PL/1 program to report on the status of customer accounts. Read 3 customer records, each containing:

CUST_NUM 999
AMT_DUE 999.99
DAYS-PAST-DUE 99

For each customer print the customer number, amount due and number of days account is overdue. Then triple space and print the total amount due and total number of days past due. Also use appropriate page and column headers.
CHAPTER 5 TEST QUESTIONS

SHORT ANSWER

1. List the following operators from highest priority to lowest priority.

   \(+, \& , <, *, :, **\)

2. The following compound condition appears in a PL/1 program:
   
   If \((W < 0 \& X < 0) \& (Y > 0 \& Z > 0)\) THEN
   
   Will this compound condition be TRUE or FALSE when:
   \(W = 5, X = 3, Y = -1, Z = 4\)?

3. Write the PL/1 statement to move a BINARY ‘1’ to EOF_FLAG when end-of-file is encountered.

MULTIPLE CHOICE

1. Which of the following are not valid PL/1 comparison operators?

   (a) <
   (b) <=
   (c) =
   (d) =

2. A header record is:

   (a) the last record of a file specifying how many data records preceded it.
   (b) the first record of a file.
   (c) the first record of a file specifying how many data records are to follow.

3. All of the following are computational conditions except:

   (a) END-OF-FILE
   (b) UNDERFLOW
   (c) ZERODIVIDE
   (d) FIXEDOVERFLOW

SHORT PROBLEMS

1. Write the PL/1 statements for the flowchart segments below. DO NOT worry about the rest of the program.

   A) [Flowchart 5.1.A]
   B) [Flowchart 5.1.B]

2. Write the PL/1 code SEGMENT to perform the following paragraph. (code just the segment, do not declare anything)

   Two numbers A and B are to be compared, if A is greater than B calculate \(C = 3 \times A\), otherwise
calculate \( C = \frac{4}{8} \). Print the result.

3. Write the PL/1 code to read a value \( N \) that is equal to the number of values of \( X \) following. Read and print all the values of \( X \).

LONGER PROBLEMS

1. Write the PL/1 program to read, print and calculate the sum of 50 integers. Print the sum when finished.

2. Sunny Valley Hotel needs to calculate how many single and double occupancy rooms have been reserved. Write a PL/1 program that reads an unknown number of records containing:

   Customer Phone # col 1-7
   Room Type col 8       D = Double
                       S = Single

Print each record and the total number of single and total number of double rooms reserved.
CHAPTER 6 TEST QUESTIONS

SHORT ANSWER

1. Define the following terms:
   (a) accuracy
   (b) reliability
   (c) readability
   (d) maintainability

2. What is the difference between a syntax error and a logic error?

3. List the two debugging aids suggested by the authors of your textbook and describe how each is used.

MULTIPLE CHOICE

1. All of the following are ways to increase the readability of a program, except:
   (a) hard coding.
   (b) indentation.
   (c) documentation.
   (d) meaningful variable names.

2. A control character in column one is used for:
   (a) documentation.
   (b) the beginning of the procedure statement.
   (c) setting the vertical spacing of the listing of source code.
   (d) setting the vertical spacing of the listing of output.

3. Which of the following did your authors suggest to include in documentation?
   (a) FUNCTION, INPUT, OUTPUT, NOTES
   (b) FUNCTION, INPUT, OUTPUT, ERRORS
   (c) STRUCTURES, INPUT, OUTPUT, ERRORS
   (d) STRUCTURES, INPUT, OUTPUT, NOTES

SHORT PROBLEMS

Given the following problems, write the comments that would precede the head of the program that would be written to solve it.

1. Jet Set Country Club needs to know how many of its members are females and between the age of 21 and 30. Each member record contains the:

   Name       col 1-20
   Age        col 21-22
   Sex        col 23
Print a report listing each females name, and age. Also print the total number of females in the above age category.

2. Your boss needs a report listing the employees in each department. Each page should start with a new department and end with the total for the department. List the name, job title, and salary for each employee. All of this information is on the employee records, along with sex, and age.
CHAPTER 7 TEST QUESTIONS

SHORT ANSWER

1. Define a case structure and give an example of one using a flowchart.

2. What is a null ELSE used for?

3. Where does control go to when a false condition is encountered in the IF-THEN?

MULTIPLE CHOICE

1. Which one of the following is not an example of a detail line?
   (a) customer name and address
   (b) price and quantity of parts ordered
   (c) total number of employees on the report
   (d) car model, color, and year

2. All of these terms are involved in report logic except:
   (a) footings
   (b) control field
   (c) comments
   (d) control break

3. What type of structure is most often used for report logic?
   (a) sequence
   (b) DO-WHILE
   (c) IF-THEN-ELSE
   (d) nested DO-WHILE

PLAYING COMPUTER

Given the following flowcharts and data, what values would be printed?

A) [Flowchart 6.A]
   Data: 20,80,0,-30,100,5,24,45,15,90,-1

B) [Flowchart 6.B]
   Data: 6666,53
   6666,94
   7878,83
   7878,68
   1212,2
   1212,200
   0003,6
   5421,48
   5421,26
SHORT PROBLEMS

Write the pseudocode for each of the flowcharts in PLAYING COMPUTER parts A) and B).

LONGER PROBLEMS

Code the PL/1 programs for each of the flowcharts in PLAYING COMPUTER part A) and B). Include appropriate page and column headers and skip to a new page for each control break in part B).
Flowchart 5.1.A

1. A > B?
   - YES: CNTA ← CNTA + 1
     - X ← A / B
     - PRINT X
   - NO: CNTB ← CNTB + 1
     - X ← B / A
Flowchart 5.1.B

1. READ X
2. EOF?
   - YES: END
   - NO: CNT ← CNT+1
3. READ X

The flowchart demonstrates a loop that reads an input until the end of file (EOF) is reached.
Flowchart 6.A

START

CTR ← 0
SUM ← 0
AVG ← 0

READ
X

X = -1?

YES

NO

PRINT
X

SUM ← SUM + X
CTR ← CTR + 1

READ
X

AVG ← SUM / CTR

PRINT
SUM, AVG

STOP
Flowchart 6.B

START

READ N

EOF?

YES

NO

TOT_QUAN ← 0

READ PART_NUM, QUAN

CUR_PART_NUM ← PART_NUM

NEW PART_NUM OR EOF

YES

NO

PRINT RECORD

TOT_QUAN ← TOT_QUAN + QUAN

READ PART_NUM, QUAN

PRINT TOT_QUAN
Flowchart 6.B Continued

TOT_QUAN ← 0

STOP