Block II.

PREPARE THE MAN

STUDENT WORKBOOK 2OSR0123-3

C-180-BUIC-WB

Computer Systems Department

BASIC PROGRAMMING PROBLEMS

(BUIC III)

January 1968

Keesler Technical Training Center
Keesler Air Force Base, Mississippi

Designed For ATC Course Use

Computer Systems Department
Keesler AFB, Mississippi

Student Workbook
January 1968

BASIC PROGRAMMING PROBLEMS

This workbook provides a guide of student exercises to be performed in Block II of Course 20SR0123-3, BUIC III Computer Programming.

CONTENTS

1. GIVEN: Values A and B (A, B integers $> \emptyset$).
WAF to satisfy the following conditional statements:

   a. If A $<$ B, then set ANS equal to A + B and stop.

   b. If A = B, then set ANS equal to 2A + 2B and stop.

   c. If A $>$ B, then set ANS equal to A - B and stop.

2. GIVEN: Values E and G. (E, G integers $> \emptyset$).
WAF to satisfy the following conditional statements:

   a. If E = 1 and G = 1, then set BAB equal to E^2G/G^2E.

   b. If E $> 3\emptyset$ and G $>$ E, then set BAB equal to E(G + 3).

   c. For all other conditions stop.

3. GIVEN: Values A and B.
WAF to satisfy the following conditional statements:

   a. If the absolute value of A is equal to or greater than the absolute value of B, set DAD equal to A + 2B.

   b. If the absolute value of B is greater than the absolute value of A, then set DAD equal to 2A + 2B.

4. GIVEN: Values X, Y. (X, Y integers $> \emptyset$).
WAF to satisfy each of the following conditional statements:

   a. If X $\geq$ Y, then set CUP equal to X/Y.

   b. If X $<$ Y, then set CUP equal to Y/X.

5. GIVEN: Values A, B.
WAF to satisfy the following conditional statements:

   a. If $A^2 + B^2 > \emptyset$, then set item VALUE = 1 and stop.

   b. If $A^2 + B^2 = \emptyset$, then set item VALUE = $\emptyset$ and stop.

6. GIVEN: Values X, Y, Z, (X, Y, Z integers $> \emptyset$).
WAF to satisfy the following conditional statements:

   a. If X = Y and X $<$ Z, then set PEN equal to Y^{15}/X^9 and stop.

   b. If X $>$ Y and X = Z, then set PEN equal to Y + X(X-Y) and stop.

c. If X $<$ Y and X $>$ Z, then set PEN equal to X^9/Y^{15} and stop.

d. For all other conditions turn on error lite #3 and stop.

7. GIVEN: D, E, F integers $>$ Ø.
WAF to satisfy the following conditional statements:

a. If D $<$ F and F $<$ E, then set ROM equal to D^3G^3.

b. If D $<$ F and F $>$ E, then set ROM equal to DG.

c. If D $>$ F and F $<$ E, then set ROM equal to D^2G.

d. If D = F and F = E, then set ROM equal to DG^2.

e. For all other conditions turn on error lite #2 and stop.

8. GIVEN: P, Q, R integers $>$ Ø.
WAF to satisfy the following conditional statements:

a. If P = Q and Q $<$ R, then set ANS1 equal to $A^2 + B^2$.

b. If P = Q and Q = R, then set ANS2 equal to $A^2 + B^2$.

c. If (P $<$ Q or R $<$ Q) and Q = 3Ø, then set ANS3 equal to $A^2/B^{1/4}$.

d. For all other conditions set ERROR to 1 and stop. (ERROR initially should be cleared.)

9. Given 5 values AØ, A1, A2, A3, A4.
WAF to satisfy the following conditional statement:

a. If their sum is equal to or less than 100, and their product less than or equal to 20^5, then set ANSWER equal to zero. 1024

b. If their sum is greater than 2^{10} and the product minus the sum is less than 100, then set ANSWER equal to one.

c. Under any other condition print out the word ERROR on the line printer.

10. GIVEN: A, B, C, D, E, Z Ø $\leq$ Z $\leq$ 5.
WAF to satisfy the following conditional statements:

a. If Z = Ø, then compute A^2 + BE + DB + DA + E^2 and store the result in item BAB.

b. If Z = 1, then compute B^3 + CD + AE + AC + D^2 and store the result in item BAB.

11. GIVEN: A, B, C, D, E, Z.
WAF to satisfy the following conditional statements:

a. If $Z < \emptyset$, then compute $A^2 + B^2/(C + B) + D + E$ and store the result in item CAD.

b. If $Z > \emptyset$, then compute $(A^2 + B^2)/(CB) + (B + D)/(B + E)$ and store the result in item CAD.

c. If $Z = \emptyset$, then compute $(C + D + A)/(D^2 + ABC)$ and store the result in item CAD.

12. GIVEN: A, B, C, D.
WAF to satisfy the following conditional statements:

a. If $A > C$, $B > D$ turn on condition light #1.

b. If $A > C$, $B < D$ turn on condition light #2.

c. If $A > C$, $B = D$ turn on condition light #3.

d. If $A < C$, $B > D$ turn on condition light #4.

e. If $A < C$, $B < D$ turn on condition light #1 and #2.

f. If $A < C$, $B = D$ turn on condition light #1 and #3.

g. If $A = C$, $B > D$ turn on condition light #1 and #4.

h. If $A = C$, $B = D$ turn on condition light #1, #2, and #3.

i. If $A = C$, $B < D$ turn on condition light #2 and #4.

13. WAF for a program to select the proper key to open a door. There are three keys of different size named KUNO, KDOS, and KTRES. The largest key will open the door.

14. WAF to examine the three values in items NEIN, NZWEI, and NDREI. Determine which is the middle value and store that item's value in MNUM. All values are positive and none are equal.

15. GIVEN: A, B, C.
WAF to compute XX the real and positive ($\geq \emptyset$) roots of the general equation $AZ^2 + BZ + C = \emptyset$ and store them in items $ROOT^1$ and $ROOT^2$.

The roots are obtained with the following formula: $XX = \dfrac{-B \pm \sqrt{B^2 - 4AC}}{2A}$

$ROOT^1$ equals -B+ "the quantity" etc.; $ROOT^2$ equals -B- "the quantity" etc.

16. King Farouk will send his aide to the market to buy 30 camels. He wants to give the aide a flow chart, which will insure that he picks satisfactory camels. Farouk wants healthy, two-humped camels, with strong legs and satisfactory saddles. Draw a flow for his aide.

17. You are the buyer for two shoe stores. One is a high class specialty store and the other is a high volume store selling shoes at low prices. Both stores have set prices on their shoes and demand to be supplied with shoes that they can sell at a 100% profit. The specialty store has authorized you to buy $5000.00 worth of shoes which are in current styles, of narrow width, and of high quality. The volume store has authorized you to buy $7500.00 worth of any type shoe. You must keep a total of the money spent on each type shoe. Draw a flow of the procedure you will follow.

18. GIVEN: An infinite number of women.

WAF to find a total of 67 women who fit into the following 6 categories. Keep a count of the number of women in each category.

a. Blond
5' 3'' - 5' 9''
Has blue eyes
Likes champagne

a.' Blond
5'3'' - 5'9''
Has blue eyes and does not like champagne OR has green eyes and likes martinis

b. Red head
5'2'' - 5'4''
Green eyes and likes sweep vermouth

b.' Red head
5'2'' - 5'4''
Has green eyes and doesn't like sweet vermouth OR has brown eyes and likes cognac

c. Brunette
5'4'' - 5'6''
Blue eyes
Likes manhattans

c.' Brunette
5'4'' - 5'6''
Has blue eyes and doesn't like manhattans OR has brown eyes and likes beer

19. Write declarations for each of these items:

a. COP contains integral values between 5 - $5\emptyset_{(1\emptyset)}$.

b. KID contains integral values between $\emptyset$ - 255.

c. POP contains a constant 294/1176, precise to 1/16.

d. JAN contains fractional values less than or equal to 1/8, precise to 1/256.

e. HOL contains maximum of 7 Hollerith characteris.

f. GIRL describes the possible statuses of 4 girls.

g. SWITCH describes the 2 possible statuses of a light switch.

h. TRAFLITE describes the 3 possible statuses of a traffic light.

i. PRO contains maximum of 3 Hollerith characters.

20. GIVEN:

ITEM PBAL	1	2Ø	S	~~12.ØØ~~	11.Ø5
ITEM DEP	1	17	S	~~ØØ.ØØ~~	Ø8.Ø5
ITEM AMT1	1	17	S	~~ØØ.ØØ~~	Ø8.Ø5
ITEM AMT2	1	17	S	~~ØØ.ØØ~~	Ø8.Ø5
ITEM NBAL	1	2Ø	S	~~12.ØØ~~	11.Ø5

The present bank balance is in PBAL, the amount deposited in DEP and the two amounts paid out in AMT1 and AMT2.

WAF to calculate the new balance and store the result in NBAL.

21. With the gross pay amount stored in item GPAY and the amounts of deductions for bonds, hospitalization insurance, and union dues in items BOND, HOSP, and UND respectively, calculate the net pay and store it in item NPAY. Declare all items.

22. GIVEN:

ITEM PLANE	1	1	V	V(BOMBER)	V(TANKER)	
ITEM BTYPE	2	2	V	V(B-70)	V(B-58)	V(B-52)
ITEM TTYPE	4	2	V	V(KC-97)	V(KC-135)	V(KB-50)
ITEM SPEED	6	2	V	V(FAST)	V(MED)	V(SLOW)

REQ'D: WAF to accomplish the following.

a. If PLANE equals BOMBER and BTYPE is a B-70, set item SPEED to FAST and stop. If BTYPE is equal to other than B-70, set SPEED to MED and stop. If PLANE does not equal BOMBER, set SPEED to SLOW and stop.

b. Determine whether PLANE equals TANKER or BOMBER. If BOMBER, check BTYPE.

If BTYPE equals B-52, set SPEED to SLOW and stop.
If BTYPE equals other than B-52, set SPEED to FAST and stop.

If PLANE is TANKER, check TTYPE

If TTYPE equals KC-97 or KC-135, set SPEED to MED and stop.
If TTYPE is KB-50, set SPEED to SLOW and stop.

23. GIVEN:

ITEM	CIV	1	2	V	V(GS-9)	V(GS-11)	V(GS-12)	
ITEM	MIL	3	2	V	V(LT)	V(CAPT)	V(MAJ)	
ITEM	WORK	5	2	V	V(NEVER)	V(RARELY)	V(OCCAS)	V(ALWAYS)
ITEM	HOURS	7	1	V	V(LONG)	V(SHORT)		
ITEM	PLAN	8	2	V	V(GOOD)	V(FAIR)	V(BAD)	

REQ'D: WAF to accomplish the following:

If CIV contains a status of GS-9, set WORK to RARELY and HOURS to SHORT and stop. If CIV equals other than GS-9, set PLAN to FAIR and check the Item MIL. If MIL equals LT, set WORK to ALWAYS, HOURS to LONG, PLAN to GOOD and stop. If MIL equals other than LT, set WORK to OCCAS and stop.

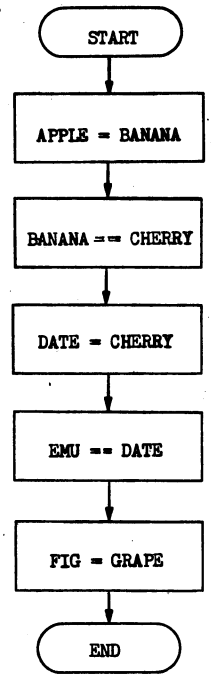
24. GIVEN:

ITEM	APPLE	1	6	U	P	Ø
ITEM	BANANA	7	6	U	P	1
ITEM	CHERRY	13	6	U	P	2
ITEM	DATE	19	6	U	P	3
ITEM	EMU	25	6	U	P	4
ITEM	FIG	31	6	U	P	5
ITEM	GRAPE	37	6	U	P	6

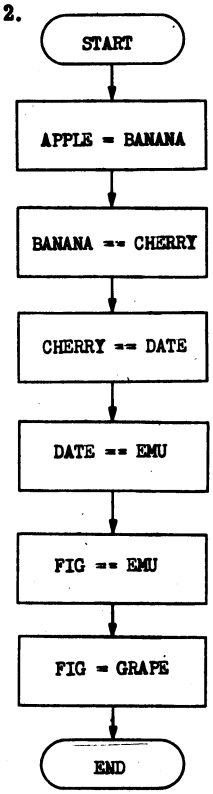
Determine what will be the contents of the items after each of the following routines is executed. Begin each routine with the original values. (P with an item declaration indicates a pre-set item.)

set A to B

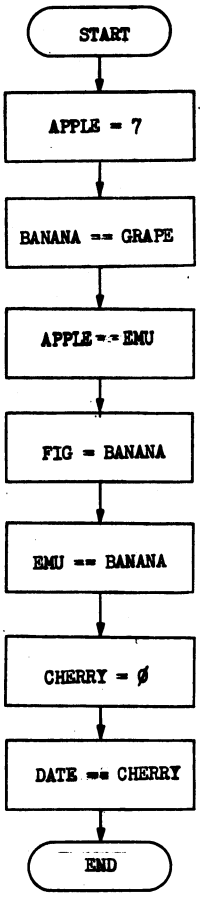
1.



2.



3.

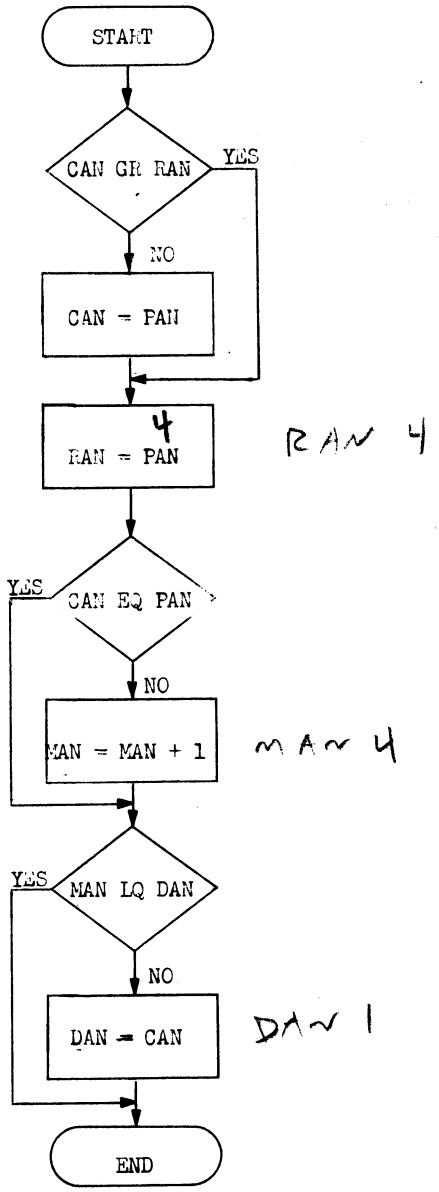


X 25. GIVEN:

ITEM	RAN	1	9	U	P	Ø
ITEM	CAN	1Ø	9	U	P	1
ITEM	DAN	19	9	U	P	2
ITEM	MAN	28	9	U	P	3
ITEM	PAN	37	9	U	P	4

Determine what will be in each item after execution of each of the following flows. Begin each flow with the original values.

1.

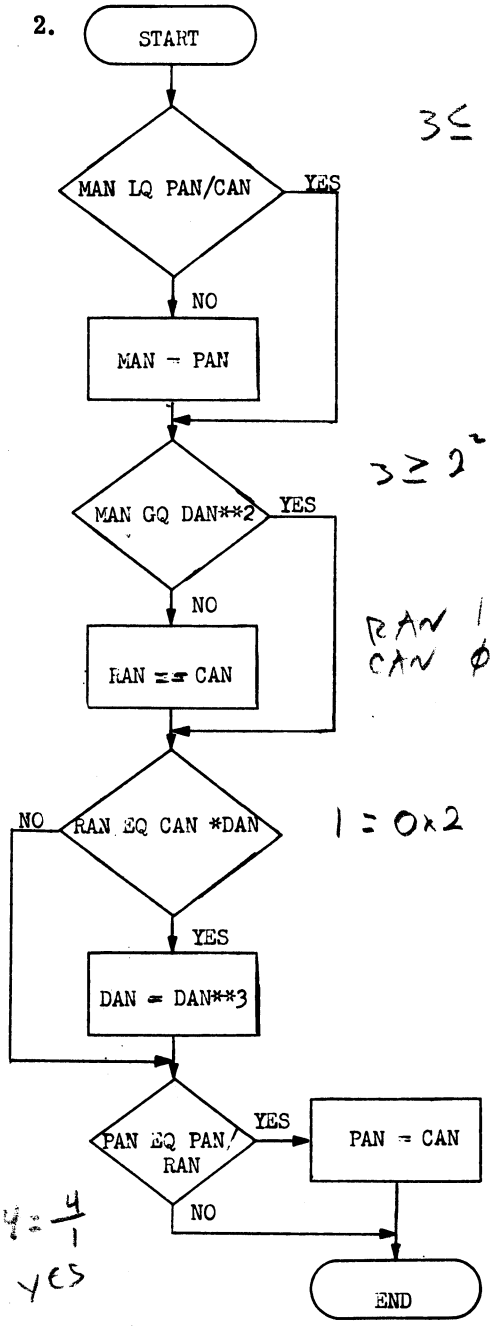


RAN 4

MAN 4

DAN 1

2.



3 ≤ 4/1 yes

3 ≥ 2² NO

RAN 1
CAN Ø

1 = 0×2 NO

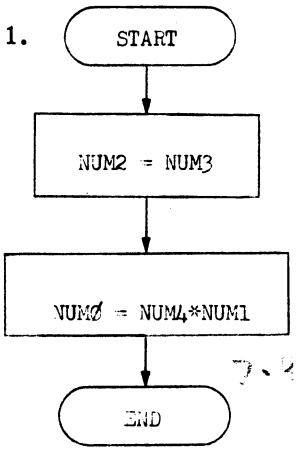
4 = 4/1
yes

26. GIVEN:

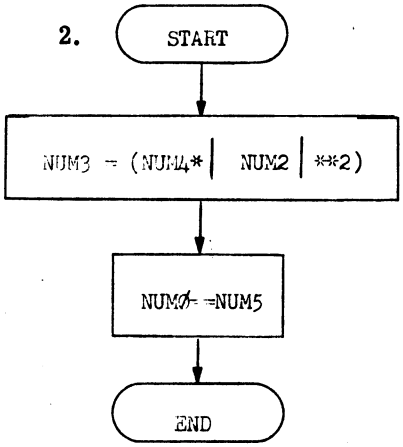
ITEM	NUMØ	1	7	S	P	2
ITEM	NUM1	8	7	S	P	4
ITEM	NUM2	15	7	S	P	-1
ITEM	NUM3	22	7	S	P	Ø
ITEM	NUM4	29	7	S	P	7
ITEM	NUM5	36	7	S	P	8

Determine the contents of each item after each of the following flows has been executed.

1.



2.



27. There is a table defined containing ten entries. In each entry is an item named FROG. WAF to do the following:

a. In entry number Ø, 1 and 2 square the value of FROG.
b. In entry number 3 increase the value in FROG by 7.
c. In entry number 4, 5, and 6 do nothing to FROG.
d. In entry number 7, 8, and 9 set FROG equal to Ø.

28. GIVEN:

TABLE	NIX	3ØØ	1	3ØØ	R	R P
ITEM	COLOR	NIXØ1	1	3	V	V(RED) V(WHITE) V(BLUE) V(GREEN) V(YELLOW) V(BLACK) V(PURPLE)
ITEM	NUMB	NIXØ1	4	3	U	
ITEM	ODD	NIXØ1	7	1	V	V(YES) V(NO)
ITEM	EVEN	NIXØ1	8	1	V	V(YES) V(NO)
ITEM	USOFA	NIXØ1	9	1	V	V(YES) V(NO)

WAF that will check the item NUMB and COLOR in each entry.

If NUMB is an even number (Ø is an even number), set the item EVEN to YES and the item ODD to NO in that entry. If the item NUMB contains an odd number, set EVEN to NO and ODD to YES. If COLOR is red, white, or blue, then set the ITEM USOFA to YES in that entry. If COLOR is equal to any other color, then set USOFA to NO in that entry.

29. GIVEN:
```
TABLE  LOOK      3ØØ  1 3ØØ  R R P
 ITEM  GIRL      LOOKØ1  1  2  V  V(PRETTY) V(FAIR) V(HOMELY)
 ITEM  FOLLOW    LOOKØ1  3  2  V  V(YES)    V(NO)   V(MAYBE)
 ITEM  VIEW      LOOKØ1  5  2  V  V(GOOD)   V(SO-SO) V(BAD)
ITEM CT  1  12  4
```

WAF that will interrogate the item GIRL in each entry. If GIRL equals PRETTY, set item VIEW to GOOD in that entry. If GIRL is not PRETTY, no additional action is required in that entry. Stop when all girls have been interrogated. Use item CT as an index word.

30. There are 20 numbers in table FISH. There are 20 different numbers in table COD. Given is table HALIBUT which is empty. Compute $FISHØ1_{(N)} + CODØ1_{(N)}$ and store the result in $HALIBUTØ1_{(N)}$. (N) indicates the entry number.

31.
```
TABLE  BEAR      1ØØØ  1  1ØØØ  R V P
 ITEM  KODIAK    BEARØ1   1  6  U
```
```
TABLE  COLOR 1ØØØ  1  1ØØØ  R V P
 ITEM  BROWN COLORØ1  1  6  U
```

WAF to store the items KODIAK into table COLOR in the same order as they appear in table BEAR. Table COLOR is empty.

32.
```
TABLE   ANIMAL   1ØØ  1  1ØØ   R V P
 ITEM   APE      ANIMALØ1  1  8  U
 ITEM   CAT→     ANIMALØ1  9  8  U
 ITEM   DOG      ANIMALØ1 17  8  U
```

a. WAF that will set CAT to Ø in every entry that APE is equal to 2.

b. WAF that will (a) set APE equal to the square of CAT and (b) set CAT to Ø in all the entries in which DOG is equal to 7. If DOG is not equal to 7, set both APE and CAT to 1. (c) In all entries, set DOG equal to Ø after it has been checked for a value of 7 before proceeding to the next entry.

33.
```
ITEM   CTR    1    1 1   U
TABLE  DATA   1ØØ   1   1ØØ   R V P
 ITEM  NUMB   DATAØ1   1  24   U
```

WAF to determine how many NUMB are less than or equal to the value 60_{10}. Store the count in CTR.

34.
```
TABLE   TRIP    19   1   19   R V P
 ITEM   MILES    TRIPØ1   1   15   U
ITEM    DIST    1   15   U
ITEM    SUM     1   3Ø   U
```
```
TABLE   TIRED  2Ø  1  2Ø  R R P
 ITEM   WEARY TIREDØ1  1  3Ø  U
```

WAF to add all values of MILES greater than the value in DIST. Store the sum in the 20th entry of table TIRED and in the non-tabular SUM.

35. GIVEN:

```
TABLE   ZERO    1ØØ  1  1ØØ     R V P
 ITEM   RADI    ZEROØ1  1   15   U

TABLE   ONE     2ØØ  1  2ØØ     R V P
 ITEM   DIAM    ONEØ1   1   15   U

ITEM    XFER    1   8   U
ITEM    VOID    1  15   U
ITEM    PI      1  15   U
```

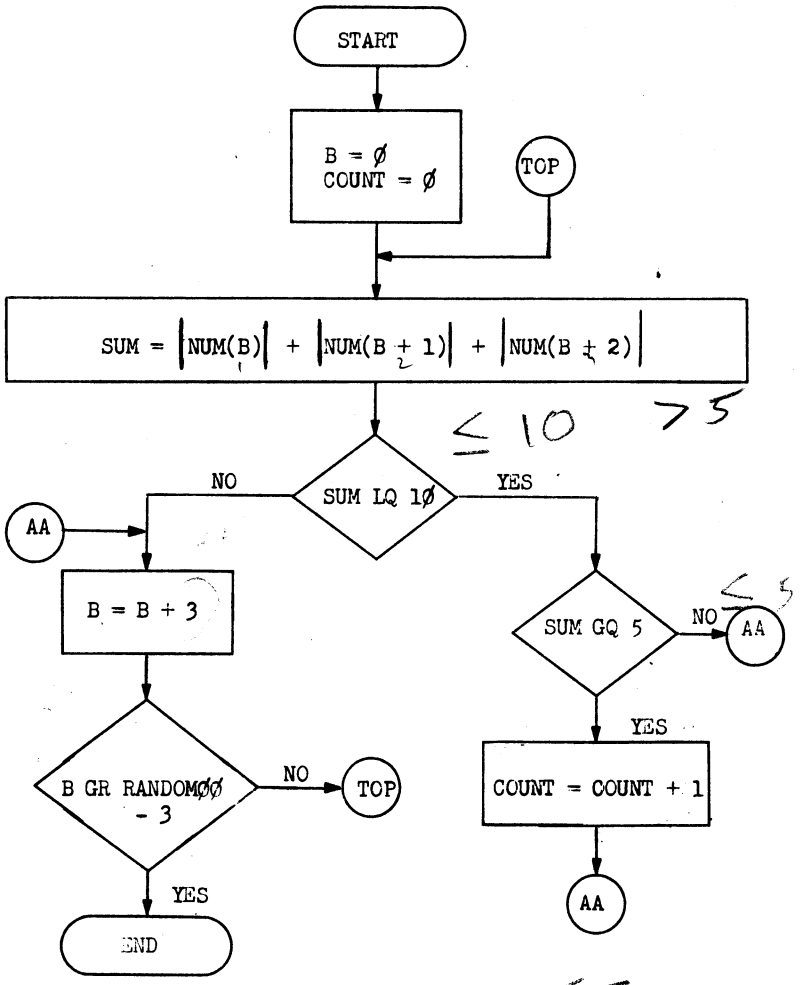
WAF to store all the values of RADI which are less than PI in table ONE starting with the first empty entry of table ONE. Keep count in XFER how many items were added to table ONE. Store the constant VOID in RADI where RADI is less than PI. Update all control items.

36. GIVEN:

```
TABLE   RANDOM    123  1  123   R V P
 ITEM   NUM       RANDOMØ1  1  48   F
ITEM    COUNT     1   11   U
ITEM    SUM       1   2Ø   S
```

Slot = 1 register 1 register = 25 bits

DETERMINE: 1. What does this flow accomplish? 2. What is the purpose of COUNT?



37. GIVEN:
```
ITEM    SCORE    1    9    U
TABLE   BOWL    12    1   12    R R P
 ITEM   PIN1     BOWLØ1       1   5  U
 ITEM   PIN2     BOWLØ1       6   5  U
```

Table BOWL, a 12_{10} entry table, which contains the number of pins knocked down in a bowling game. The item PIN 1 contains the number of pins knocked down by the first ball of that frame and the item PIN2 contains the number of pins knocked down by the second ball (if a second ball is needed). Entry Ø of the table represents frame 1, and entry 1 represents frame 2, ... entry 1Ø represents the two bonus balls awarded for a mark in the tenth frame. (If 2 bonus balls are awarded and the first ball is a strike, the pin count from the second ball will be in PIN1 of BOWL(11).

WAF to calculate the total score for the game and store this value in SCORE. (NOTE: This game has already been bowled.)

38. GIVEN:
```
TABLE   DATA    365    1   365    R R P
 ITEM   INFO    DATAØ1      1 2Ø   U
 ITEM   DAY     DATAØ1     21 1Ø   U
 ITEM   WDHD    DATAØ1     31  1   B
```

To facilitate its accounting procedures, a department store numbers the days of the year from 1 to 365. Accounting data is stored in memory in item INFO. The item DAY contains the day number. The item WDHD indicates whether the day is a working day or holiday. If WDHD = Ø, a working day is indicated; if WDHD = 1, a holiday is indicated. Initially all WDHD = Ø.

WAF which will place the correct value into WDHD. January 1st for this year falls on Friday. Only Saturdays and Sundays are holidays.

39. GIVEN:
```
TABLE   FORTY   4Ø   1   4Ø   R R P
```

There are 40 numbers stored in table FORTY. (slot = 1 reg.). The values may be positive or negative.

WAF to compile two totals - one of the positive values and one of the negative values. Store these totals in the non-tabular items POS and NEG respectively.

40. GIVEN:
```
TABLE    VALUES     99    1   99   R R P
 ITEM    NUM        VALUESØ1       1 11 S
ITEM     ANSWER     1     1    V   V(PLUS) V(MINUS)
```

WAF to set item ANSWER to the status of PLUS if there are more positive than negative numbers in table VALUE or the status of MINUS if there are more negative numbers. Assume there are no items with a value of zero.

41.
```
TABLE   ABC   1ØØ   1   1ØØ   R V P
 ITEM   BAKE  ABCØ1      1 8   U
```

WAF to find the largest number in table ABC and store that value in LNUMB.

42. TABLE CKPOINT 1ØØ 1 1ØØ R R P
ITEM PTID CKPOINTØ1 1 24 H
ITEM TIMREP CKPOINTØ1 25 16 D ← BCD
ITEM WHERE 1 24 H

WAF to set item WHERE to the identity of the checkpoint (PTID) that reported in nearest to 1851 hours (TIMREP). Assume there are exactly 100 checkpoints and all reports were received between 0000 to 2400 hours.

43. GIVEN: TABLE NUM 3ØØ 1 3ØØ R V P
ITEM DIGIT NUMØ1 1 12 U
ITEM FIND 1 12 U
ITEM INDX 1 12 U

The contents of the key item DIGIT increases by increments of 2. DIGIT will always be an even number. FIND may be an odd or even number. Search table NUM for the value in FIND. If it is found, set the value of FIND equal to DIGIT and set INDX to the value of the entry number in which the desired value was found. If the value is not found, set all values in NUM to odd values.

44. GIVEN: ITEM SORTED 1 1 V V(TRUE) V(FALSE)
TABLE SENIOR 3ØØ 1 3ØØ R V P
ITEM NAME SENIORØ1 1 24 H
ITEM MAJOR SENIORØ1 25 6 H
ITEM AVERAGE SENIORØ1 33 15 U

Table SENIOR contains the name, major fields of study and averages of all seniors in a given university. This table is supposed to be sorted in ascending sequence on the key item AVERAGE.

WAF to set item SORTED to TRUE if the table is sorted as specified, or to FALSE if the condition is not met.

45. GIVEN: ITEM ENTNO 1 1Ø U
ITEM ERROR 1 1 V V(YES) V(NO)
ITEM REG 1 9 U
ITEM KONST 1 3Ø U
TABLE PON 512 1 512 R V P
ITEM NUMB PONØ1 1 3Ø U

WAF, using binary search, to search table PON for the value contained in KONST. (PON is in descending order.) If the value is found, set ENTNO equal to the entry number and set ERROR equal to NO. If the value is not found, set ERROR equal to YES.

46. A non-tabular item NSD which contains a number between Ø and 32 inclusive. Table NOS contains the squared values for numbers between Ø and 32 inclusive. (Ent #Ø = Ø, Ent #1 = 1, Ent #2 = 4, Ent #3 = 9, etc.) Write a statement using Direct Look-up that will set the item ANS to the squared value of NSD.

47. GIVEN:

```
TABLE    UFD        28   1   28   R  R  P
 ITEM    FAIRSHR    UFDØ1     1   2Ø   U  13.Ø7
TABLE    OFFICER    2ØØØ 2  1ØØØ  R  V  P
 ITEM    NAME       OFFICERØ2    1 48  H
 ITEM    GRADE      OFFICERØ1    1  2  V  V(0-1) V(0-2) V(0-3) V(0-4)
 ITEM    YROSERV    OFFICERØ1    3  3  V  V(LS2) V(OV2)V(OV3) V(OV4)
                                             V(OV6) V(OV8) V(OV10)
 ITEM    AMTGIVEN   OFFICERØ1    6  3Ø U  23.Ø7
 ITEM    HALO       OFFICERØ1   36  1  V  V(ANGEL)  V(FALLEN)
ITEM     TOTFAIR    1  4Ø    U  33.Ø7
ITEM     TOTACT     1  4Ø    U  33.Ø7
ITEM     COMMANDER  1  1    V  V(HAPPY)   V(UNHAPPY)
```

Table OFFICER contains in item NAME, all the names of the officers in grade 0-1 to 0-4 in an organization. Item GRADE contains their grade and YROSERV, status corresponding to their length of service. AMTGIVEN contains the amount each officer contributed to the United Fund.

Table UFD contains the fair share for each officer in item FAIRSHR. The table is organized in ascending order by grade and within the grade structure in ascending order by years of service. (The first 7 entries are for grade 0-1. Entry #0 is for 0-1, LS2; entry #1 is for 0-1, OV2; etc.)

WAF to set the item HALO in each entry equal to ANGEL, if AMTGIVEN in that entry is greater than or equal to the officer's fair share amount. If not, set item HALO to FALLEN. Compute the total amount given by the organization and put this total into TOTACT. Also compute the total if each officer had given his fair share and put this total into TOTFAIR. If TOTACT is greater than or equal to TOTFAIR, set COMMANDER equal to HAPPY and stop. If not, set COMMANDER equal to UNHAPPY and branch to program BUST. (NOTE: An algorithm can be used to solve this problem.)

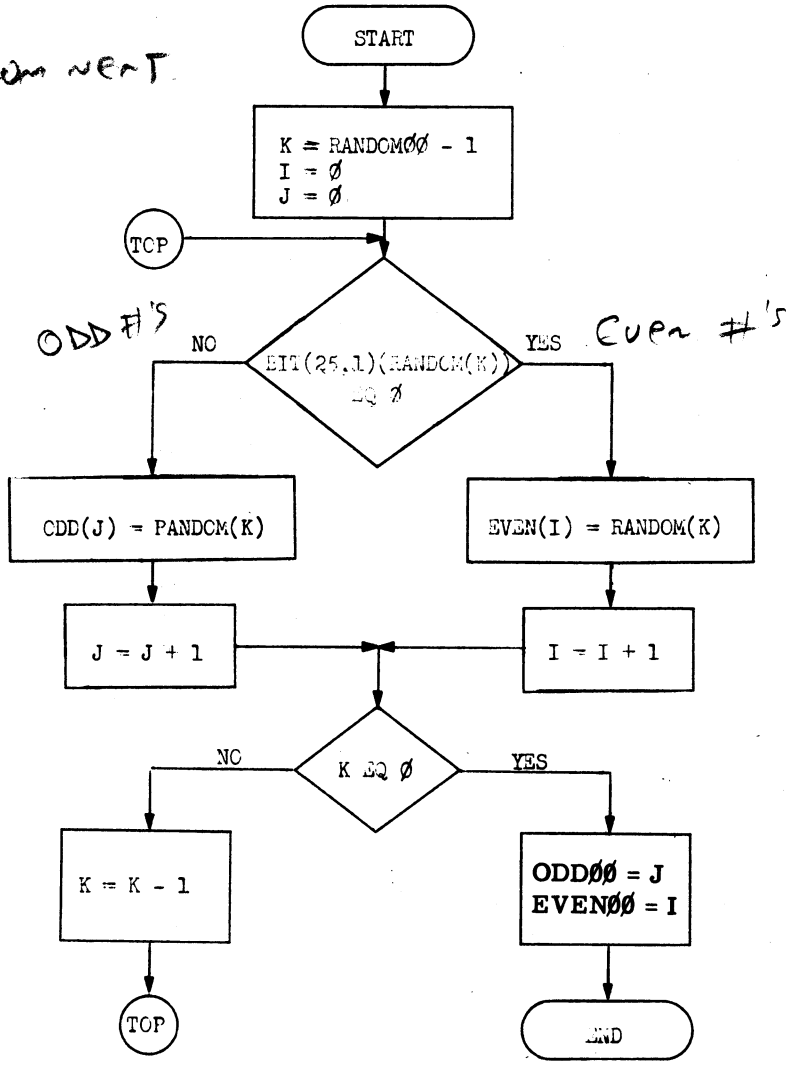
48. WAF to load index word KAT with the contents of bits #4-#8 of the item BOB. (ITEM BOB 1 12 U). Then decrement the index word by one until its value reaches Ø. (Take into account the condition that 4-8 might be all zeroes.)

49. WAF to add together the contents of bit #1-#6, #4-#9, #7-#11, #10-#12 in BOB. Place this sum into index word S.

X 50. GIVEN: Tables RANDOM, EVEN, and ODD are variable length.
Slot = 1 register Register length = 25 bits

What does this program accomplish?

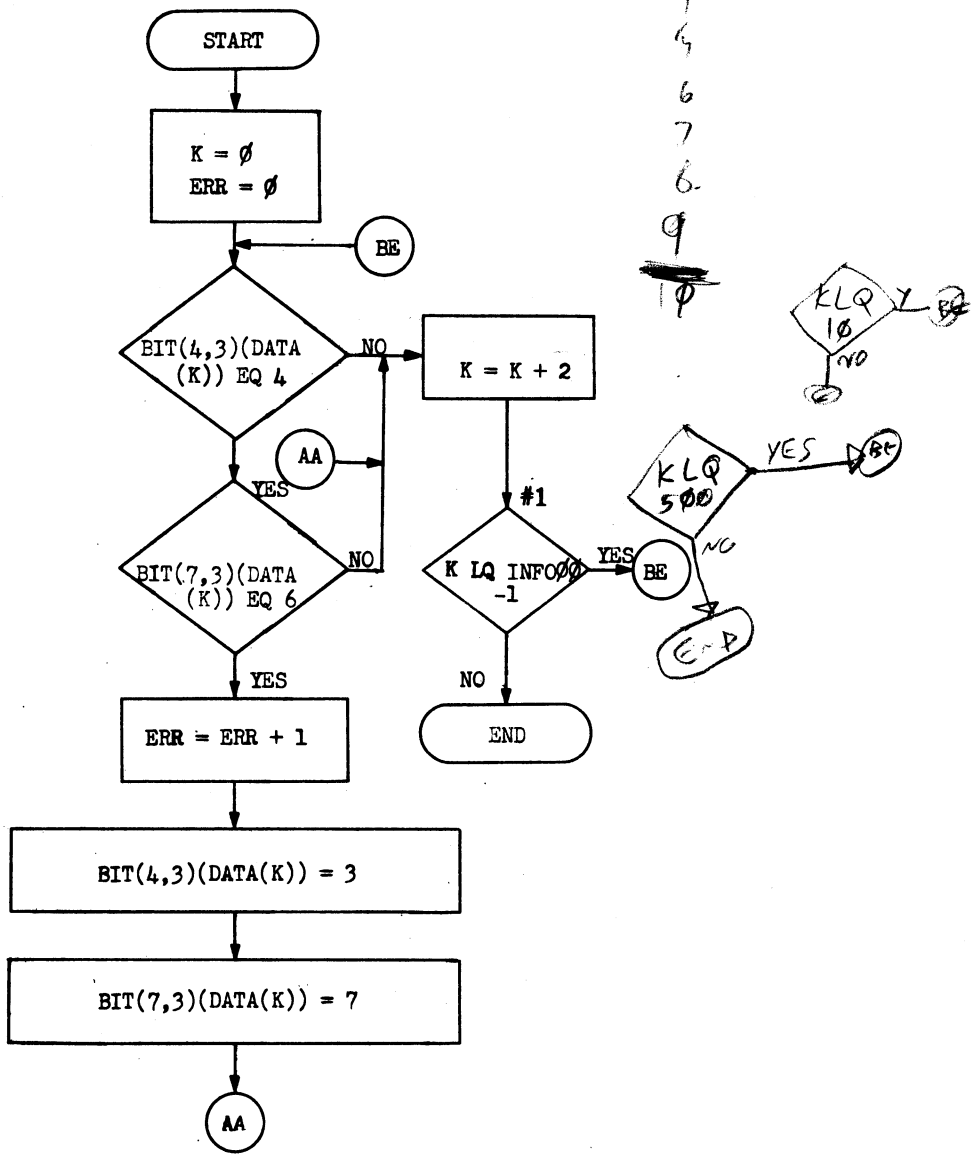
K = RANDOM next



NOTE SPACE ① Sorts table random into even & odd numbers. ② Stores Even #'s in EVEN & Odd #'s in ODD. ③ Each number now in EVEN or ODD has the same entry # it had in RANDOM. ④ Sets next of ODD + EVEN = the # of entries in each. ⑤ NO. Entry are in reverse order.

51. GIVEN:

ITEM	ERR	1	1Ø	U	
TABLE	INFO	5ØØ	1	5ØØ	R (R) P
ITEM	DATA	INFOØ1		1	1Ø U



a. What does this flow accomplish?

b. What does ERR contain at the end of this program?

c. If box #1 contained the statement, "K LQ 5ØØ", how would the flow have to be changed to accomplish the same result?

NOTE SPACE

(A) Beginning at entry Ø check every other entry of ITEM DATA. If a 4 is in bits 4,5+6 and if a 6 is found in bits 7,8+9 change the 4 to a 3 + change the 6 to a 7.

(B) counts the # of times 4's + 6's were changed to 3's+7's

(C) we go through table more than necessary

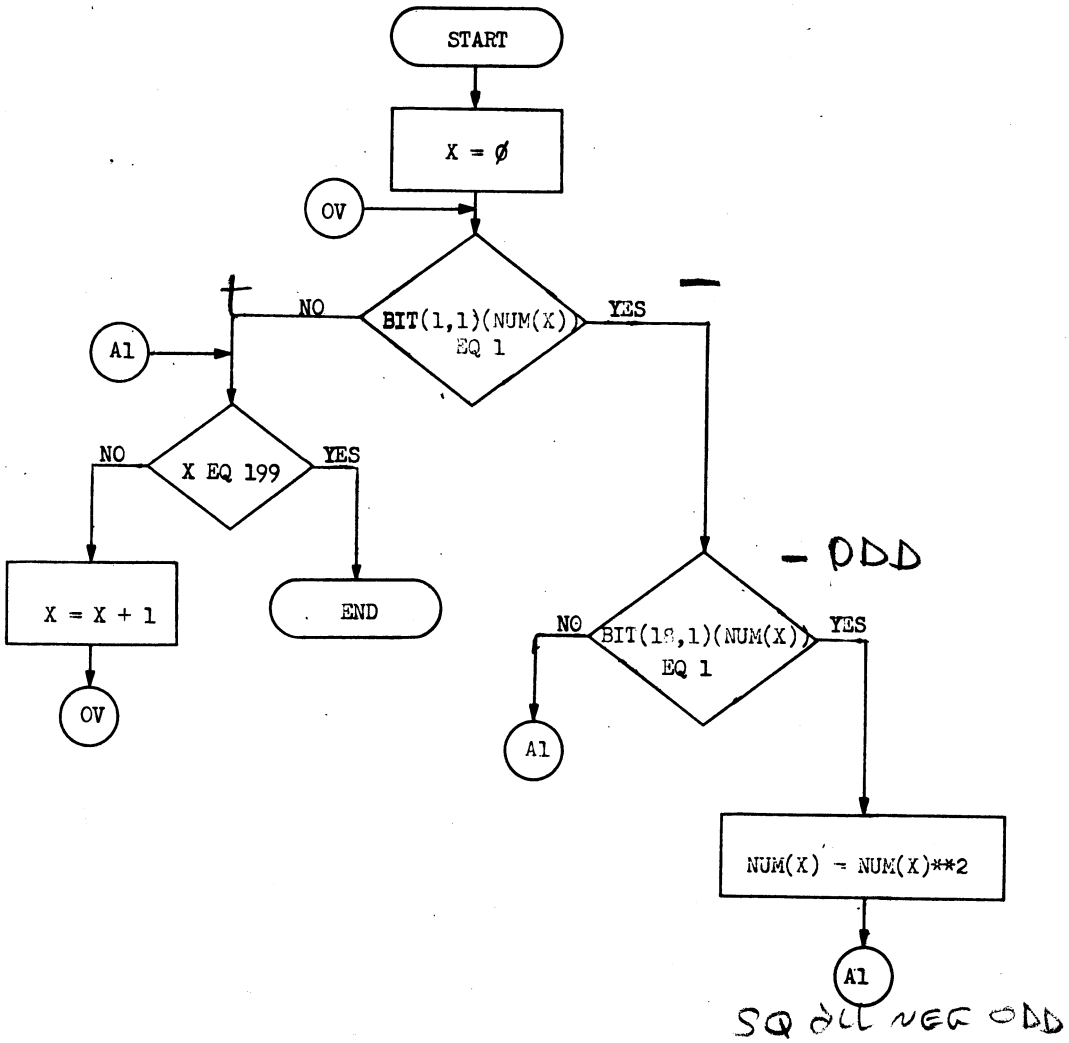
(7)

52. WAF that will branch to the program SQRT if bits #7 and #9 of item CALL equal Ø's. If they equal 1's, go to CBRT. If they don't meet either of these conditions, go to program QUADEQ.

53. WAF to count in item CTR the number of 1's in item BULB. BULB has 48 bits. If BULB has 30 or more 1's, set the first 24 bits to Ø's and the second 24 bits to 1's. If less than 30 1's, set the first 24 bits to 1's and the second 24 bits to Ø's.

54. GIVEN:

TABLE	TBL	2ØØ	1	2ØØ	R	R	P
ITEM	NUM	TBLØ1		1	18	S	



a. What does this flow do?

SQUARE ALL negative odd #'s in Table TBL

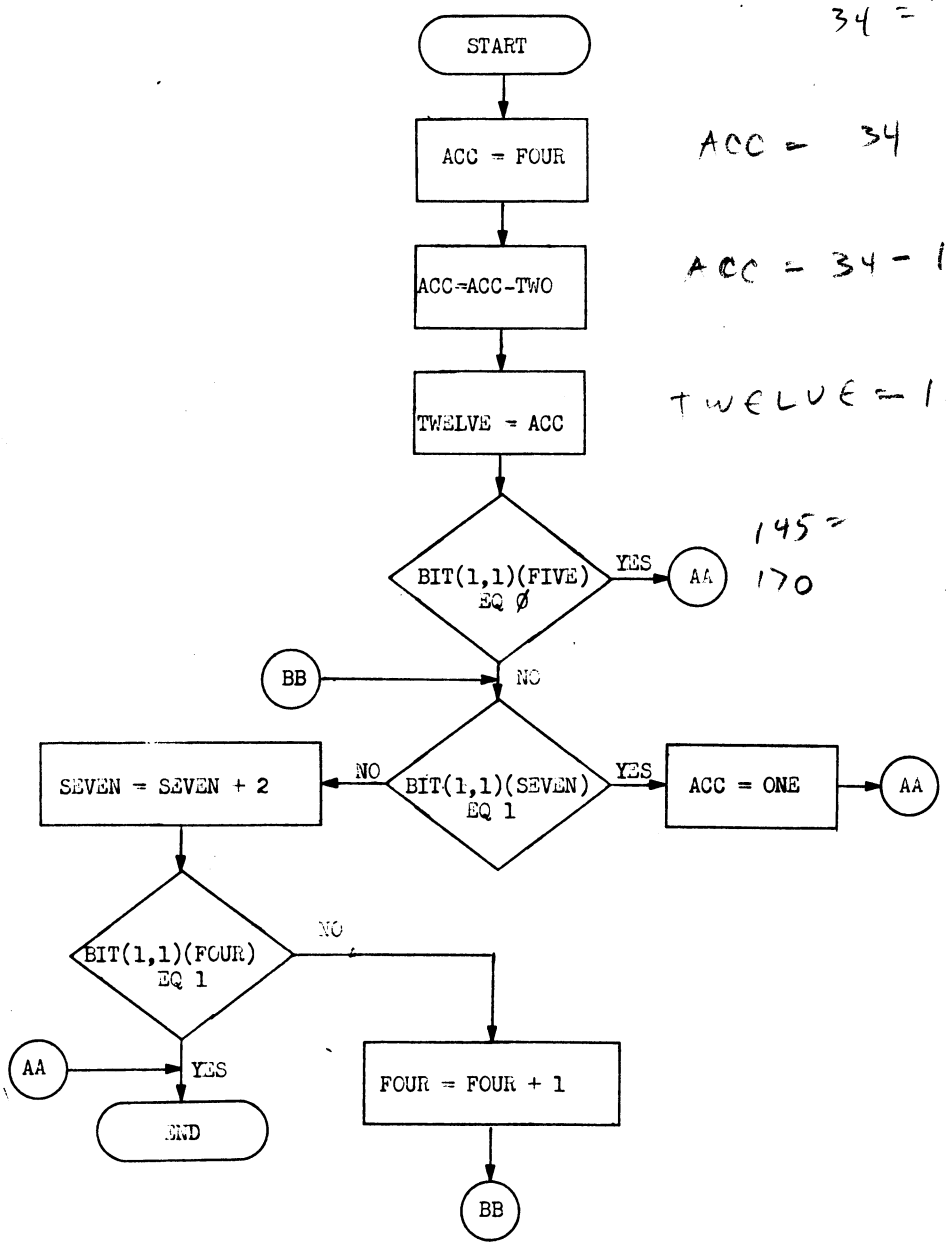
NOTE SPACE

55.

ITEM	ACC	1	9	U	P	ø 1 (ALL VALUES ARE OCTAL)
ITEM	ONE	1	3	U	P	1
ITEM	TWO	1	6	U	P	16
ITEM	FOUR	1	6	U	P	34 40
ITEM	FIVE	1	7	U	P	145
ITEM	SEVEN	1	8	U	P	170 200
ITEM	TWELVE	1	9	U	P	10 16

Determine the value of all items at the end of this flow.

NOTE: These are preset items.



NOTE SPACE

56. GIVEN:

TABLE	SCORE	1ØØØØ	1	1ØØØØ	R	V	P
ITEM	MARK	SCOREØ1		1 8	S		
ITEM	QUES	SCOREØ1		Ø 4Ø	U		

Table SCORE contains the results of a 4Ø-question examination given to a maximum of 1Ø,ØØØ people.

Each bit of item QUES corresponds to one of the 4Ø questions where a setting of 1 means the question was answered correctly, and Ø, incorrectly answered.

The weighting of the test is:

QUESTION	VALUE OF EACH QUESTION
1-8	1/2 point
9-3Ø	1 point
31-36	5 points
37-4Ø	10 points

WAF to set each entry of item MARK to the numeric grade of each student and then exit to program EXEC.

57. Given a computer clock of 48 bits that is pulsed "n" times every second. When program ADP starts the contents of the clock are extracted and saved in item CLOCK1. When ADP is finished operating, the contents of the clock are extracted and saved in item CLOCK2. WAF to determine if ADP took longer than 5 seconds to operate. The value 5 seconds is stored as a constant in MAXT. If the program took longer than 5 seconds to operate, turn on condition lite #4 and print out the actual time to operate on the line printer. If the program operated in 5 seconds or less than 5 seconds, branch to program BCDP.

58. There are 10 numbers, not consecutive, but in ascending order in table NOS. Arrange these numbers in descending order in the same location. (slot = 1 reg.)

59. Table DECK contains positive and negative numbers in random sequence. WAF to sort the absolute values in table DECK in descending order. CARD is the key item in DECK.

60.

TABLE	POS	5ØØ	1	5ØØ	R	V	P
ITEM	NUMB	POSØ1		1 1Ø	U		

Positive numbers are stored in random order in POS. WAF to store these numbers in ascending order in table POS.

61. TABLE RECEIVER 5ØØ 1 5ØØ R V P
ITEM MSGE RECEIVERØ1 1 48 H

Table RECEIVER can contain up to 500 messages, each message consisting of up to 56 characters. The last character in each message is a period. (Periods may be embedded within the message.) WAF to set item NOTUSED to the total number of unused bytes of the entries containing messages, and set item LONGEST to the number of characters representing the longest message. (A register = 8 characters)

62.

TABLE	PERS	8ØØØ	1	8ØØØ	R	V	P	TABLE	PERSX	5Ø	1	5Ø	R	V	P
ITEM	SERIAL	PERSØ1	1	8	U			ITEM	SERIALX	PERSXØ1	1	8	U		
ITEM	RANK	PERSØ1	9	18	H			ITEM	RANKX	PERSXØ1	9	18	H		
ITEM	DEPENDENTS	PERSØ1	27	6	U			ITEM	DEPENDENTSX	PERSXØ1	27	6	U		
ITEM	CLEARANCE	PERSØ1	33	6	H			ITEM	CLEARANCEX	PERSXØ1	33	6	U		
ITEM	MONTHS	PERSØ1	39	8	U	Ø4.Ø4		ITEM	MONTHSX	PERSXØ1	39	8	U	Ø4.Ø4	

(All items in table PERSX are identical to the items in table PERS.) Table PERS contains information concerning members of the AF assigned to a certain station. Table PERSX contains information concerning up to 50 new members who have transferred to this station. Both tables are in ascending order. Key items - SERIAL, SERIALX. WAF to merge/insert the data from PERSX into PERS maintaining the original order.

63. TABLE RESULTS 2ØØØØ 2 1ØØØØ R V P
ITEM SERIAL RESULTSØ1 1 48 H
ITEM AGE RESULTSØ2 1 7 U
ITEM IQ RESULTSØ2 8 12 U

Table RESULTS contains the serial number, age, and IQ of up to 10,000 members of the armed services. WAF to delete those entries with an IQ of less than 90 and then repack the table.

64. Given variable length tables ONE and TWO. The key item for ONE is UNO. The key item for TWO is DOS. Each is sorted in descending order. Each has positive and duplicate numbers. WAF to merge ONE and TWO into table THREE, which will be in ascending order.

65. Given table MEMORY of $10{,}000_8$ registers. (slot = 1 reg) An area of MEMORY that is to be cleared is indicated in the following manner: all of core between the address of "67_8" and "77_8" inclusive, as indicated by bits #1-#6 of each register. Search through MEMORY for these indicators. When they are found, put the addresses of the area to be cleared in the input parameters BEGIN and END. Then call in SBR CLEAR. Take into consideration the fact that no area may be cleared or that more than 1 area may be cleared.

66. Write the subroutine to clear MEMORY indicated in the preceding problem. Use the parameters given.

67. This will be a drum maintenance program to check information transfer lines and drum storage capability.

```
TABLE   CAB   2ØØØ   1   2ØØØ   R V P
TABLE   RUG   2ØØØ   1   2ØØØ   R V P
```

WAF to transfer the data from table CAB onto drum channel #26. When the transfer is complete read the information from drum channel #26 into table RUG. Then do a comparison entry by entry between the 2 tables. If there are any errors, save the entry numbers in table ERROR and program an error stop. If there are no errors branch to program TAPE MAINT.

```
68. ITEM    CONST    1    7    U            TABLE   XYZ   4ØØ  1  4ØØ  R V P
    TABLE   ABC     4ØØ   1   4ØØ  R V P    ITEM    PRADR XYZØ1    1 15 S
     ITEM   RADR    ABCØ1      1  15   S
     ITEM   PROC    ABCØ1      16 1    B
```

The above data is stored on drum channel #7. Raw radar returns are stored in RADR and are constantly being processed. WAF to determine if RADR should be processed or cleared. If the PROC equals 1, then the return has been processed and that entry requires clearing. If PROC equals Ø, process RADR by multiplying RADR by CONST. Store the processed radar return in item PRADR and set PROC to a value of 1. Before branching to program TRACK write table ABC onto drum #2, drum location MNP.

```
69. TABLE    BIRD        2ØØ  1   2ØØ   R  R  P
     ITEM    AARDVARK  BIRDØ1   1    5  U
     ITEM    DODO      BIRDØ1   6    5  U
     ITEM    CRANE     BIRDØ1   11   5  U
     ITEM    EGG       BIRDØ1   16   5  U
```

WAF to determine the largest item in each entry, including EGG. If AARDVARK is the largest, set EGG of that entry equal to "1". If DODO is the largest, set EGG equal to "2". If CRANE is the largest, set EGG equal to "3". (None are equal). Before branching to program FRAME, write table BIRD on to drum #1, drum location TWEET, and clear drum #3.

```
70. ITEM    CONST   1    7    U
    TABLE   RRTN    4ØØ   1    4ØØ   R  V  P
     ITEM   RETRN   RRTNØ1       1      5  S
```

Raw radar returns are stored in table RRTN. This table is stored on drum #2, drum location DRT. WAF to read this table into core and operate on it in the following manner - multiply each of the returns by CONST and store the result in its original location. Before branching to program TRK VEL write this table on drum #2, drum location DRT.

71. GIVEN: A card reader with 101 cards in the card hopper. The first 100 cards contain the height (in feet) of 100 triangles. The 101st card contains the common base (in inches) of these 100 triangles. WAF to read the card reader and calculate the area in sq. ft. (A = 1/2bh). The 100 heights will be read into table HGT. The base will become a non-tabular item BASE. Store the computed areas in table AREA. Before branching to program MORT, write the table HGT and AREA as the 2nd and 3rd record on TD #1.

72.															
	TABLE	BAT	5ØØ	1	5ØØ	R	V	P	INPUT & OUTPUT PARAMETER ITEMS						
	ITEM	ROB	BATØ1		1 12	U				ITEM	SIDE1	1	12	U	
	ITEM	RAY	BATØ1		13 12	U				ITEM	SIDE2	13	12	U	
	ITEM	ALPHA	BATØ1		25 24	U				ITEM	SIDE3	25	24	U	

WAF to sort table BAT into descending sequence based on ROB. After table BAT has been sorted call in subroutine PYTHAGORUS'S THEOREM. Before using the SBR set SIDE1 = ROB and SIDE2 = RAY. The result of the computation will be placed in SIDE3. Place this result in ALPHA. Cycle thru the entire table performing this computation for every entry.

SUPPLEMENTAL PROGRAMMING PROBLEMS

PROBLEM 1

You are required to construct a flow to solve the following problem:

Total the cost of five Christmas purchases. This total will not exceed $100.00. The cost of each item will be contained in items PUR1 thru PUR5.

You will subtract the amount of the total bill, item TBIL, from $100.00 and determine the amount of change to be received.

The change will be broken down into the following denominations:
$20.00; $10.00; $5.00; $1.00; $.50; $.25; $.10; $.05; $.01.

You will determine the number of each denomination that will be received as change. The items for the number of denominations received as change will be TWEN, TEN, FIVE, ONE, PFIVE, PTIV, PTEN, POFIVE, POONE.

20. 10. 5. 1.
.50 .25 .10 .05 .01

Write the declarations necessary for any items you use.

(will be handed in)

NOTE SPACE

Value V(

PROBLEM 2

Construct a flow to solve the following problem:

Change the structure of TABLE BAD from parallel to serial.

```
TABLE   BAD  3ØØ  3  1ØØ  R  V  P
 ITEM    BONE      BADØ1     1  48  U
 ITEM    BTWO      BADØ2     1  48  U
 ITEM    BTRE      BADØ3     1  48  U
```

Declare and explain any necessary item you use.

NOTE SPACE

PROBLEM 3

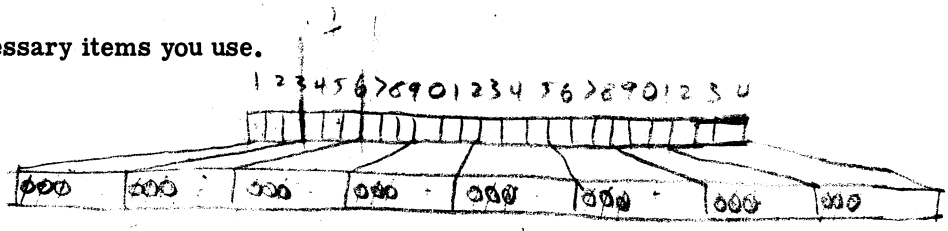
You are required to construct a flow to solve the following problem:

Convert a binary number into Hollerith octal. The binary number is contained in item BNUM which is an unsigned integer of 24 bits. It is located in bits #1-#24 of the register, the rest of which is blank. Store the answer in item HONUM, which is a Hollerith item of 48 bits occupying an entire register.

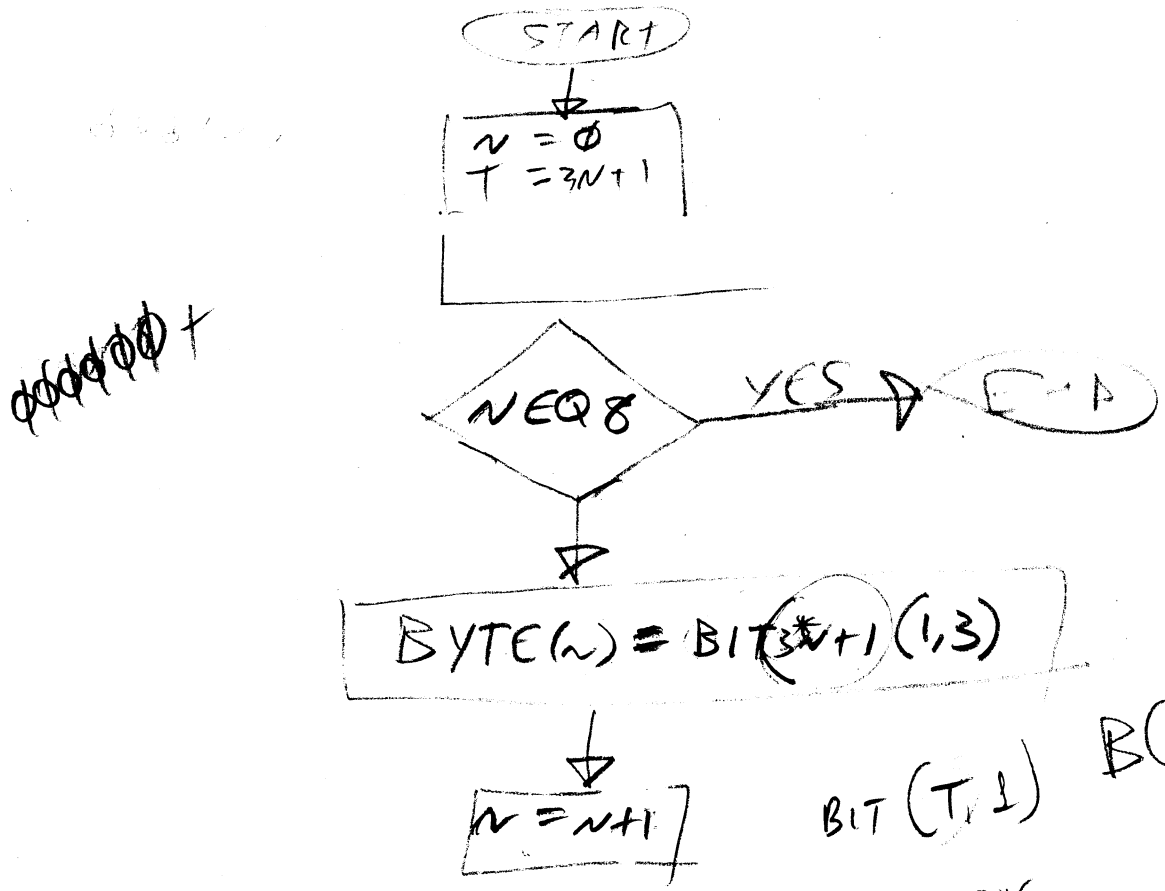
Declare and explain any necessary items you use.

NOTE SPACE

6 110



Byte(N) = Bit(N)



BYTE(N) = 000000 + BIT(T,3)

PROBLEM 4

You are required to construct a flow to solve the following problem:

Convert a binary number into Hollerith decimal. The binary number is contained in item BNUM which is an unsigned integer that is right justified in the register and whose maximum value is less than 100. Store the answer in item HDNUM, which is left justified in a register. Declare and explain any necessary items you use.

```
ITEM  NUM     42   7   U   Ø7.ØØ
ITEM  HDNUM   1        12    H
```

NOTE SPACE

PROBLEM 5

Construct a flow to solve the following problem:

Table TSD contains track numbers. They are in random sequence. Sort them using the bucket sort on the basis of the track number, item TTRN. Refer to the section of the study guide for the operation of the bucket sort.

Using the following declarations:

```
TABLE STN 1ØØ  1  1ØØ  R  V  P
  ITEM SDLI STNØ1  1  12  U
  ITEM STRN STNØ1  31  18  T
TABLE LNK 1Ø  1  1Ø  R  R  P
  ITEM LINK LNKØ1  48  U
TABLE TSD 1ØØ  1  1ØØ  R  V  P
  ITEM TTRN TSDØ1  1  18  T
```

NOTE SPACE

Computer Systems Department

Character	12 Bit Hol	6 Bit Hol
0	0,	01 0000
1	,1	00 0001
2	,2	00 0010
3	,3	00 0011
4	,4	00 0100
5	,5	00 0101
6	,6	00 0110
7	,7	00 0111
8	,8	00 1000
9	,9	00 1001
BLANK		00 0000
A	12,1	11 0001
B	12,2	11 0010
C	12,3	11 0011
D	12,4	11 0100
E	12,5	11 0101
F	12,6	11 0110
G	12,7	11 0111
H	12,8	11 1000
I	12,9	11 1001
J	11,1	10 0001
K	11,2	10 0010

Character	12 Bit Hol	6 Bit Hol
L	11, 3	10 0011
M	11, 4	10 0100
N	11, 5	10 0101
O	11, 6	10 0110
P	11, 7	10 0111
Q	11, 8	10 1000
R	11, 9	10 1001
S	0, 2	01 0010
T	0, 3	01 0011
U	0, 4	01 0100
V	0, 5	01 0101
W	0, 6	01 0110
X	0, 7	01 0111
Y	0, 8	01 1000
Z	0, 9	01 1001
$	11, 8-3	10 1011
+	, 8-3	00 1011
-	, 8-4	00 1100
,	0, 8-3	01 1011
*	11, 8-4	10 1100
.	12, 8-3	11 1011

TABLE OF POWERS OF 2

2^n	n	2^{-n}
1	0	1.0
2	1	0.5
4	2	0.25
8	3	0.125
16	4	0.062 5
32	5	0.031 25
64	6	0.015 625
128	7	0.007 812 5
256	8	0.003 906 25
512	9	0.001 953 125
1 024	10	0.000 976 562 5
2 048	11	0.000 488 281 25
4 096	12	0.000 244 140 625
8 192	13	0.000 122 070 312 5
16 384	14	0.000 061 035 156 25
32 768	15	0.000 030 517 578 125
65 536	16	0.000 015 258 789 062 5
131 072	17	0.000 007 629 394 531 25
262 144	18	0.000 003 814 697 265 625
524 288	19	0.000 001 907 348 632 812 5
1 048 576	20	0.000 000 953 674 316 406 25
2 097 152	21	0.000 000 476 837 158 203 125
4 194 304	22	0.000 000 238 418 579 101 562 5
8 388 608	23	0.000 000 119 209 289 550 781 25
16 777 216	24	0.000 000 059 604 644 775 390 625
33 554 432	25	0.000 000 029 802 322 387 695 312 5
67 108 864	26	0.000 000 014 901 161 193 847 656 25
134 217 728	27	0.000 000 007 450 580 596 923 828 125
268 435 456	28	0.000 000 003 725 290 298 461 914 062 5
536 870 912	29	0.000 000 001 862 645 149 230 957 031 25
1 073 741 824	30	0.000 000 000 931 322 574 615 478 515 625
2 147 483 648	31	0.000 000 000 465 661 287 307 739 257 812 5
4 294 967 296	32	0.000 000 000 232 830 643 653 869 628 906 25
8 589 934 592	33	0.000 000 000 116 415 321 826 934 814 453 125
17 179 869 184	34	0.000 000 000 058 207 660 913 467 407 226 562 5
34 359 738 368	35	0.000 000 000 029 103 830 456 733 703 613 281 25
68 719 476 736	36	0.000 000 000 014 551 915 228 366 851 806 640 625
137 438 953 472	37	0.000 000 000 007 275 957 614 183 425 903 320 312 5
274 877 906 944	38	0.000 000 000 003 637 978 807 091 712 951 660 156 25
549 755 813 888	39	0.000 000 000 001 818 989 403 545 856 475 830 078 125

NOTES

SAVE A LIFE

If you observe an accident involving electrical shock,
DON'T JUST STAND THERE - DO SOMETHING!

RESCUE OF SHOCK VICTIM

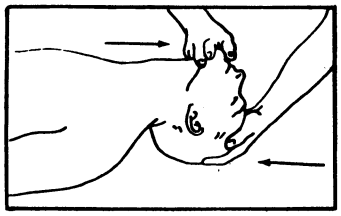
The victim of electrical shock is dependent upon you to give him prompt first aid. Observe these precautions:

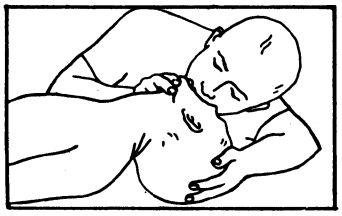
1. Shut off the high voltage.
2. If the high voltage cannot be turned off without delay, free the victim from the live conductor. REMEMBER:
   a. Protect yourself with dry insulating material.
   b. Use a dry board, your belt, dry clothing, or other non-conducting material to free the victim. When possible PUSH - DO NOT PULL the victim free of the high voltage source.
   c. DO NOT touch the victim with your bare hands until the high voltage circuit is broken.

FIRST AID

The two most likely results of electrical shock are: bodily injury from falling, and cessation of breathing. While doctors and pulmotors are being sent for, DO THESE THINGS:

1. Control bleeding by use of pressure or a tourniquet.
2. Begin IMMEDIATELY to use artificial respiration if the victim is not breathing or is breathing poorly:
   a. Turn the victim on his back.
   b. Clean the mouth, nose, and throat. (If they appear clean, start artificial respiration immediately. If foreign matter is present, wipe it away quickly with a cloth or your fingers).





   c. Place the victim's head in the "sword-swallowing" position. (Place the head as far back as possible so that the front of the neck is stretched).
   d. Hold the lower jaw up. (Insert your thumb between the victim's teeth at the midline - pull the lower jaw forcefully outward so that the lower teeth are further forward than the upper teeth. Hold the jaw in this position as long as the victim is unconscious).
   e. Close the victim's nose. (Compress the nose between your thumb and forefinger).
   f. Blow air into the victim's lungs. (Take a deep breath and cover the victim's open mouth with your open mouth, making the contact air-tight. Blow until the chest rises. If the chest does not rise when you blow, improve the position of the victim's air passageway, and blow more forcefully. Blow <u>forcefully</u> into <u>adults</u>, and <u>gently</u> into <u>children</u>.
   g. Let air out of the victim's lungs. (After the chest rises, quickly separate lip contact with the victim allowing him to exhale).
   h. Repeat steps f. and g. at the rate of 12 to 20 times per minute. Continue rhythmically without interruption until the victim starts breathing or is pronounced dead. (A smooth rhythm is desirable, but split-second timing is not essential).

DON'T JUST STAND THERE - DO SOMETHING!