

TCEA
HIGH SCHOOL
PROGRAMMING CONTEST

STATE PROBLEM SET
APRIL 27, 2002

**Texas Computer Education Association
2002 High School Programming Contest
State Problem Set**

0.0 Extremes (practice problem)

2.1 Currency Converter

2.2 Matching Parentheses?

2.3 Letters Triangle

2.4 Lawn Supplies

2.5 Is The Weather Normal?

2.6 How Many Times?

5.1 Time Passes

5.2 Diamond Word

5.3 Print Commands

5.4 Card Value

5.5 Decode It

5.6 Binary Operators

9.1 Alphabetical Numbers

9.2 Spiral Unwind

9.3 Millionaire

9.4 Olympics Ratings

9.5 Concordance

9.6 Family Tree

Problem 2.1 Currency Converter

General Statement: Convert a given US dollar amount to Pounds, Lira, Francs, Marks, or Yen.

Input: The first line in the data set is an integer that represents the number of data collections that follow. Each collection consists of an integer US dollar amount followed by the name of the currency to be converted to.

Name of Data File: pr21.dat

Output: Output a dollar sign (\$), the US dollar amount and the words CONVERTS TO and then the converted amount followed by the currency unit name.

If the conversion factor is an integer, output an integer. If it is a decimal, round to 2 decimal places.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: The US dollar amount is an integer in the range 1..500.

All letters are upper case.

Discussion: The conversion factors for \$1 are: 0.84 POUNDS
2040 LIRA
9.85 FRANCS
3.23 MARKS
260 YEN

Sample Input: 3
17 POUNDS
25 MARKS
228 YEN

Sample Output: \$17 CONVERTS TO 14.28 POUNDS
\$25 CONVERTS TO 80.75 MARKS
\$228 CONVERTS TO 59280 YEN

Problem 2.2 Matching Parentheses?

General Statement: Read a string and determine whether each left parenthesis '(' has a matching right parenthesis ')'.

Input: The first line of the data set for this problem is an integer that represents the number of strings that follow. Each string is on a separate line.

Name of Data File: pr22.dat

Output: Output either MATCH or NO MATCH.

All letters are upper case.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

(and) are the only enclosure symbols in the string.

Discussion: The left parenthesis must come before the right parenthesis in order to count as a matched set of parentheses.

Sample Input: 3
 (3 + (7 * 2) - 6)
 HELLO AND (WELCOME (TO THE) TCEA (CONTEST)
 TODAY) IS ((SATURDAY())

Sample Output: MATCH
 NO MATCH
 NO MATCH

Problem 2.3 Letters Triangle

General Statement: For an input letter, output the alphabet from A through that letter in a triangle pattern. The number of letters on a line is the same as the line number of the pattern.

Input: There are an unknown number of values on a single line. An asterisk (*) is used to indicate the end of the data set.

Name of Data File: pr23.dat

Output: All letters are upper case.

 The columns are to be aligned vertically.

 Include some white space between outputs.

 The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

Discussion: If the last line of the pattern is incomplete, fill in each “empty” space with an asterisk (*).

Sample Input: DJC*

Sample Output: A
 BC
 D* *

 A
 BC
 DEF
 GHIJ

 A
 BC

Problem 2.4 Lawn Supplies

General Statement: Calculate the number of bags of fertilizer, lime, and grass seed needed based on the square footage of the lawn to be treated and seeded.

Input: All data is on a single line. The first integer indicates how many integers follow.

Name of Data File: pr24.dat

Output: For each lawn, output the number of bags of fertilizer, the number of bags of lime, and the number of bags of grass seed – in that order. Output the number of bags, followed by the word BAG(S) and then the item type.

All letters are upper case.
Include some white space between outputs.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: The area value is in the range 1..30000.

Discussion: You must buy whole bags, and you cannot buy less than the square footage requires for coverage.

Coverage Factors: 50# bag fertilizer covers 5000 square feet
40# bag of lime covers 2000 square feet
5# bag of grass seed covers 1000 square feet

Sample Input: 3 13000 4575 10000

Sample Output: 3 BAG(S) FERTILIZER
7 BAG(S) LIME
13 BAG(S) SEED

1 BAG(S) FERTILIZER
3 BAG(S) LIME
5 BAG(S) SEED

2 BAG(S) FERTILIZER
5 BAG(S) LIME
10 BAG(S) SEED

Problem 2.5 Is The Weather Normal?

General Statement: Given the actual high and low temperatures for the day and the normal high and low temperatures for that day, calculate the average difference from normal.

Input: The first line of the data set for this problem is an integer that represents the number of data sets that follow. Each data set is on a separate line and consists of today's high, today's low, normal high, and normal low – in that order.

Name of Data File: pr25.dat

Output: If the average difference is negative, do not output the negative sign (-).

Output the amount of deviation from normal, followed by the words DEGREE(S) ABOVE NORMAL, or by the words DEGREE(S) BELOW NORMAL.

Round to 1 decimal place. A trailing zero is required if the average is an integer.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: Temperatures are in the range –15..100 degrees.

The average temperature difference will not be zero.

Discussion: Determine the average of the difference of the high temperatures and the difference of the low temperatures.

Sample Input:

```
3
75 45 78 40
50 32 45 30
56 48 62 45
```

Sample Output:

```
1.0 DEGREE (S) ABOVE NORMAL
3.5 DEGREE (S) ABOVE NORMAL
1.5 DEGREE (S) BELOW NORMAL
```

Problem 2.6 How Many Times?

General Statement: For an input string, determine how many times an input letter occurs in that string.

Input: There are 2 lines of data. The first line is the input string. The second line contains an unknown number of letters on a single line. An asterisk (*) is used to indicate the end of the data set.

Name of Data File: pr26.dat

Output: Output the letter, followed by an equal sign (=) and then the count for that letter.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

The maximum string length is 60.

Sample Input: ALL IS QUIET NOW, BUT WAIT!
QAET*

Sample Output: Q = 1
A = 2
E = 1
T = 3

Problem 5.1 Time Passes

General Statement: Read 2 times (based on a 24-hour clock) and determine how much time has elapsed from time #1 to time #2.

Input: The first line of the data set for this problem is an integer that represents the number of data sets that follow. Each data set is on a separate line and consists of 2 times in the form hh:mm:ss. Values less than 10 have a leading zero.

Name of Data File: pr51.dat

Output: Output the number of hours, the number of minutes, and the number of seconds that have passed from the first to the second time.

Label each time unit
All letters are upper case.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: This is a 24-hour clock.

Hours are in the range 0..23. Minutes and seconds are in the range 0..59.

Discussion: If time #2 is "before" time #1, it is the following day.

Sample Input:

```
3
10:03:43 15:00:58
07:25:00 20:52:14
16:30:30 05:40:10
```

Sample Output:

```
4 HOURS 57 MINUTES 15 SECONDS
13 HOURS 27 MINUTES 14 SECONDS
13 HOURS 9 MINUTES 40 SECONDS
```

Problem 5.2 Diamond Word

(PAGE 1 OF 2)

- General Statement:** Read a word and output that word in a diamond pattern such that the entire word is the center horizontal line of the diamond and each row out from this center drops a letter on each end of the word.
- Input:** The first line of the data set for this problem is an integer that represents the number of strings that follow. Each string is on a separate line.
- Name of Data File:** pr52.dat
- Output:** Output the diamond pattern using all upper case letters.
- The output is to be formatted exactly like that for the sample output given below.
- Assumptions:** All letters are upper case.
- The maximum word length is 10.
- Discussion:** If there are an odd number of letters, the top and bottom of the diamond will be the middle letter. If there are an even number of letters, the top and bottom of the diamond will be the middle 2 letters.

See page 2 for sample run.

Problem 5.2 Diamond Word

(PAGE 2 OF 2)

Sample Input: 3
 SAMPLE
 PROGRAM
 PROBLEMS

Sample Output: MP
 AMPL
 SAMPLE
 AMPL
 MP

 G
 OGR
 ROGRA
 PROGRAM
 ROGRA
 OGR
 G

 BL
 OBLE
 ROBLEM
 PROBLEMS
 ROBLEM
 OBLE
 BL

Problem 5.3 Print Commands

(PAGE 1 OF 2)

General Statement: Read a string and output the results of the commands FIRST, BUT FIRST, LAST, and BUT LAST.

Input: The first line of the data set for this problem is an integer that represents the number of strings that follow. Each string is on a separate line.

Name of Data File: pr53.dat

Output: Output is to be in 2 columns. The first column is the commands in order FIRST, BUT FIRST, LAST, BUT LAST. The second column in the result of the command. Each column is to be left-aligned with some white space between columns.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

The string contains only letters and 1 space between words.

The maximum string length is 80.

The maximum word length is 12.

The maximum number of words is 6.

Discussion:	<u>Command</u>	<u>Word</u>	<u>Sentence</u>
	FIRST	1 st letter	1 st word
	BUT FIRST	omit 1 st letter	omit 1 st word
	LAST	last letter	last word
	BUT LAST	omit last letter	omit last word

See page 2 for sample run.

Problem 5.3 Print Commands (PAGE 2 OF 2)

Sample Input: 3
 VISIT US ON THE WEB
 ADDRESS
 EAGLE PASS IS IN TEXAS

Sample Output: FIRST VISIT
 BUT FIRST US ON THE WEB
 LAST WEB
 BUT LAST VISIT US ON THE

 FIRST A
 BUT FIRST DDRESS
 LAST S
 BUT LAST ADDRES

 FIRST EAGLE
 BUT FIRST PASS IS IN TEXAS
 LAST TEXAS
 BUT LAST EAGLE PASS IS IN

Problem 5.4 Card Value

(PAGE 1 OF 2)

General Statement: Read a collection of playing cards and compute the total numerical value of that "hand".

Input: The first line of the data set for this problem is an integer that represents the number of strings that follow. Each string is on a separate line.

The card value is first and the suit is second. There is an underscore () between "cards".

Name of Data File: pr54.dat

Output: Output the calculated sum followed by the word POINTS.

The output is to be formatted exactly like that for the sample output given below.

Assumptions All letters are upper case.

The maximum number of cards in a hand is 10.

Each card occurs only 1 time in a hand.

Discussion: The suits are: C = Clubs
D = Diamonds
H = Hearts
S = Spades

Card values are: 2..10 = face value
A = 11
J, Q, K = 10

If only 2 suits are in the hand, add 10 points.

If all cards are of the same single suit, add 25 points.

See page 2 for sample run.

Problem 5.4 Card Value

(PAGE 2 OF 2)

Sample Input:

```
3
2C_10H_AD_KH_4S
9H_QD_QC_3H_3D_AC_AH
7C_8C_AC_4D
```

Sample Output:

```
37 POINTS
57 POINTS
40 POINTS
```

Problem 5.5 Decode It

General Statement: Read a letter sequence that was used to encode a secret message. The first letter replaced each Z, the second letter replaced each Y, and so on. Then read the encoded message and decode it.

Input: The first line of the data set for this problem is an integer that represents the number of data sets that follow. Each data set consists of 2 lines. Line #1 is the encoding list, and line #2 is the secret message.

Name of Data File: pr55.dat

Output: All letters are upper case.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

The maximum string length is 80.

Any non-letter characters were not encoded.

Sample Input: 3
NMPOLKRQJITSHGVUFEXWDCZYBA
GCCR GC EH RXC WMG
NMOPQRLKJISTUVHGFEXYZDCBA
PWT'K AYJAED TY NZBJAK?
MNLOKPJQIRHSGTFUEVDWCXBYAZ
1313 TSYUVGWAVJB FZGX

Sample Output: MEET ME IN THE GYM
WHO'S AFRAID OF ZEBRAS?
1313 MOCKINGBIRD LANE

Problem 5.6 Binary Operations

General Statement: Read 2 lists of integer elements. Determine the results of the binary operations AND, OR, XOR, and NOR.

Input: The first line of data is list A, and the second line is list B. There are an unknown number of integers in each list. A -1 is used to indicate the end of each list. The next line is an integer that represents the number of operators that follow. Each operator is on a separate line.

Name of Data File: pr56.dat

Output: Output A, followed by the operator, followed by B =, and then the result with these numbers in ascending order. If an answer is "empty", output the word NULL.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: The master set is the range 0..10.

Each input list contains at least 1 element. There are no duplicates within a list.

Discussion: AND yields all elements occurring in both lists.
OR yields all elements occurring in either list.
XOR yields all elements occurring in either but not both lists.
NOR yields all elements in the master set not in either list.

Sample Input: 5 6 9 3 8 -1
2 3 1 5 4 9 6 8 -1
4
AND
OR
NOR
XOR

Sample Output: A AND B = 3 5 6 8 9
A OR B = 1 2 3 4 5 6 8 9
A NOR B = 0 7
A XOR B = 1 2 4

Problem 9.1 Alphabetical Numbers

General Statement: Given a collection of integers, sort them alphabetically based on the words for these numbers.

Input: The first line in the data set is an integer that represents the number of data collections that follow. There are an unknown number of integers in each data collection. The integer 0 is used to indicate the end of the collection of integers.

Name of Data File: pr91.dat

Output: List the “sorted” integers in each collection on a separate line.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: The maximum number of integers is 10.

The integers are in the range 1..19.

There are no duplicates within a data set.

Discussion: 1 2 3 4 would be sorted as 4 1 3 2. Four, one, three, and two are alphabetical.

Sample Input: 3
7 5 12 9 16 0
4 14 19 6 10 3 2 0
11 12 13 14 15 0

Sample Output: 5 9 7 16 12
4 14 19 6 10 3 2
11 15 14 13 12

Problem 9.2 Spiral Unwind

General Statement: Given a collection of letters that fill a square matrix, unwind in the given direction. The collection lists the rows of the matrix in order.

Input: The first line of the data set for this problem is an integer that represents the number of strings that follow. Each string is on a separate line. The first letter of the string is the unwinding direction (R = right, L=left).

Name of Data File: pr92.dat

Output: Output the list of unwound letters on a single line.

All letters are upper case.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

The maximum number of letters is 25.

There may be duplicate letters within the matrix.

Discussion: A square matrix has the same number of rows and columns.

Choose the matrix size based on the number of letters in the data set.

Fill the matrix row by row working from left to right.

Begin unwinding the matrix at the top left corner.

Sample Input: 3
RABCDEFGHIJKLMNPO
LABCDEFGHIJKLMNPO
RQRABATSCA

Sample Output: ABCDHLPONMIEFGKL
AEIMNOPLHDCBFJGK
QRATACSBA

Problem 9.3 Millionaire

General Statement: Read a set of 15 correct answers for the millionaire game. Then read the contestant's responses and determine how much money the player wins.

Input: The first line of the data set for this problem is an integer that represents the number of data sets that follow. Each data set consists of 2 lines. Line #1 is the 15 correct answers for the game, and line #2 is the list of player responses.

Name of Data File: pr93.dat

Output: Output a dollar sign (\$) followed by the amount won.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

Discussion: Question dollar values:

#1	\$100	#6	\$2000	#11	\$64000
#2	\$200	#7	\$4000	#12	\$125000
#3	\$300	#8	\$8000	#13	\$250000
#4	\$500	#9	\$16000	#14	\$500000
#5	\$1000	#10	\$32000	#15	\$1000000

After question #5, \$1000 is guaranteed. After question #10, \$32000 is guaranteed.

Sample Input: 3
ABCDABCDABCDABC
ABCDABCA
ABABABCD CDCBCB
ABABABCD CDA
ABCABCDABCABCD A
ABCABCDABCA

Sample Output: \$1000
\$32000
\$64000

Problem 9.4 Olympics Ratings

(PAGE 1 OF 2)

General Statement: Read a file containing the names and medal counts of several countries who participated in the Olympics. Rank them by the number of gold medals earned. Secondly, rank them by the total number of medal earned.

Input: The first line in the data set is an integer that represents the number of data collections that follow. Each data collection contains the 3-letter abbreviation of the country name followed by the number of gold, silver, and bronze medals – in that order.

Name of Data File: pr94.dat

Output: For output #1, output the words BY GOLD, then each line will list the place ranking followed by the country name. If there is a tie, the first country is listed beside the place, and the additional countries are listed in vertical alignment. When finished, the country column is to be left-justified.

For output #2, output the words BY TOTAL. Follow the directions of output #1 for listing the countries.

All letters are upper case.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

All country names are abbreviated to 3 letters.
The maximum number of countries is 12.

Discussion: First place is awarded to the one with the largest count of medals, second place to the second largest count, and so on.

If there is a tie, all will receive the same ranking and they are to be listed alphabetically within that rank. The next ranking awarded will be the rank as if there were no ties and each country was numbered consecutively.

See page 2 for sample run.

Problem 9.4 Olympics Ratings

(PAGE 2 OF 2)

Sample Input:

```
7
NOR 17 5 6
SWE 16 5 9
GBR 4 6 9
FRA 10 6 6
ITA 10 5 4
GER 17 6 6
AUS 18 3 2
```

Sample Output:

```
BY GOLD
1 AUS
2 GER
   NOR
4 SWE
5 FRA
   ITA
7 GBR

BY TOTAL
1 SWE
2 GER
3 NOR
4 AUS
5 FRA
6 GBR
   ITA
```

Problem 9.5 Concordance

General Statement: For a given text file, generate a list of words in alphabetical order that are in the text file and the lines on which those words occur.

Input: The first line of the data set for this problem is an integer that represents the number of lines that follow.

Name of Data File: pr95.dat

Output: Output the words in alphabetical order. After each word write the line number of each occurrence of that word. Start the number lists in the same column.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

The maximum line length is 40.

The maximum number of lines is 6.

Sample Input: 3
DOO WOP DOO WOP DOO WOP
SHOoby DOObY DOO
WOO WOO

Sample Output: DOO 1 1 1 2
DOObY 2
SHOoby 2
WOO 3 3
WOP 1 1 1

Problem 9.6 Family Tree

General Statement: Given the family relationships below, output all persons who fit the input characteristic. These characteristics are SIBLING, PARENT, CHILD, GRANDPARENT, and GRANDCHILD.

Family #1: Ann and Marty have children Bill, Cathy, and Frank.
Bill and Alice have no children.
Cathy and Don have children Matt and Sally.
Frank and Jill have child Sarah.

Family #2: Debbie and Phil have children Jill and Betty.
Jill and Frank have child Sarah.
Betty and Paul have children Mary, Jane, and Bart.

NOTE: The 2 families are connected by the marriage of Frank and Jill.

Input: The first line of the data set for this problem is an integer that represents the number of data sets that follow. The first word in the data set is the relationship, and the second is the name of the person whose relative(s) is/are to be located.

Name of Data File: pr96.dat

Output: Output the names of all who meet the relationship criterion.

All letters are upper case.

The output is to be formatted exactly like that for the sample output given below.

Assumptions: All letters are upper case.

Sample Input: 3
SIBLING BART
PARENT BETTY
GRANDCHILD MARTY

Sample Output: MARY JANE
DEBBIE PHIL
MATT SALLY SARAH