1. Find the occurrence of each unique word from given string. Accept input from user. \

e.g.: input: "Ruby is nice lang. ruby follows OO principles"

outputs

Ruby -> 2   
 is -> 1  
 nice ->1  
 lang -> 1  
 follows -> 1  
 oo -> 1  
 principles -> 1

2. If n is an integer greater than 0, n factorial (n!) is the product: n\* (n-1) \* (n-2) \* ( n-3)… \*

By convention, 0! = 1.  
 You must write a program that allows a user to enter an integer between 1 and 7.

Your program must then compute the factorial of the number entered by the user.

Your solution MUST perform a computation (i.e., you may not simply print “5040” to the screen as a literal value if the input is 7).

Example 1:  
 Enter a number: 4 ( 4! = 24)  
 Example 2:  
 Enter a number: 7 ( 7! = 5040)

3. At UAB football games, Blaze does push ups after each Blazer score. After the first Blazer touchdown (and point after), Blaze does 7 pushes ups. After the second touchdown and point after, the score is now 14 and Blaze does 14 push ups.

Write a program that calculates how many total push ups Blaze does during the whole game. Assume that only 7-point touchdowns (including the point after) occur. Prompt for the final score and print out how many push ups Blaze has done.

Example 1:  
 Enter final score: 21  
 Push ups: 42

Example 2:  
 Enter final score: 7  
 Push ups: 7

4. General Statement:

Read the amount of money you have and the prices of the items you intend to buy. Determine whether you have enough money to buy everything you selected or whether you are short of money. If you do not have enough money, indicate the amount of the shortfall. Be sure to include 8% tax when figuring the amount, you need.

**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 money amounts in each data set. The value –1 is used to indicate the end of the collection of prices.

**Output:** All letters are to be upper case. Include the amount of shortfall if you do not have enough money. This money amount is to have a dollar sign ($) in front of the amount and it is to be rounded to 2 decimal places. The output is to be formatted exactly like that for the sample output given below.

**Assumptions:** The –1 used to indicate the end of a data collection is not part of the data for the problem.

**Sample Input:** 10.50 7.60 1.26 3.49 -1  
 15.75 6.00 3.98 -1  
 21.00 5.25 5.75 4.76 3.98 1.50 -1

**Sample Output:** $2.84 SHORT  
 ENOUGH MONEY  
 $1.94 SHORT

5. General Statement:

Output a given word horizontally and multiple times vertically so that each letter in the horizontal word matches the position of that letter vertically.

**Input:** The first line in the data set is an integer that represents the number of words that follow. Each word is on a separate line.

**Output:** The horizontal word is to be in the centre of the output. Each vertical word is to read down from the top. The first vertical word uses the first letter of the horizontal word. The last vertical word uses the last letter of the horizontal word.  
 All outputs are to have the same left edge vertically.

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.

Sample Input:

3

TEST

SAMPLE

ART

Sample output:

T

TE

TES

TEST

EST

ST

T

S

SA

SAM

SAMP

SAMPL

SAMPLE

AMPLE

MPLE

PLE

LE

E

A

AR

ART

RT

T

6.

How Many Weeks?

General Statement: A town in Death Valley has a water tank that contains 10,000 gallons of water. If there is no rain, calculate the number of weeks the water will last for an input weekly water usage.

Input: The data set is on a single line. There are an unknown number of integers in the data set. The integer 0 is used to indicate the end of the data.

Output: Use all upper case letters. The output is to be formatted exactly like that for the sample output given below.

Assumptions: The weekly usage does not exceed 10,000 gallons. The 0 used to indicate the end of the data is not part of the data for the problem.

Discussion: Do not include the last week if the water remaining for that week is less than the weekly usage amount.

Sample Input: 1750 1000 4325 0

Sample Output:

1750 GALLONS PER WEEK WILL LAST 5 WEEKS

1000 GALLONS PER WEEK WILL LAST 10 WEEKS

4325 GALLONS PER WEEK WILL LAST 2 WEEKS

7.

General Statement: Read a sentence and output the count of the number of 1 letter words, the number of 2-letter words, etc.

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

Output: Do not output any counts of 0. The letter counts are to be in ascending order by the number of letters in the word. All letters are upper case.

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

Assumptions: Any punctuation in the sentence is not part of the word length. The maximum word length is 10 letters. The maximum sentence length is 70 characters.

Sample Input:

3

TODAY’S SATURDAY

HI SAM, HOW ARE YOU?

MARCH, THEN APRIL, THEN MAY

Sample Output:

COUNT OF 6 = 1

COUNT OF 8 = 1

COUNT OF 2 = 1

COUNT OF 3 = 4

COUNT OF 3 = 1

COUNT OF 4 = 2

COUNT OF 5 = 2

8.

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.

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

9.

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.

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.

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

10.

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.

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