



**BITMAP 2011**

**BINUS International Computer Championship 2011**



**ACM Based Competition**

**Problem Sets – 1<sup>st</sup> ROUND**

## PREAMBLE

Do note the following information of utmost importance regarding input and output:

- Read all input from keyboard, i.e. use ***stdin***, ***System.in***, ***cin*** or equivalent. Input will be redirected from a file to form the input to your submission.
- Do NOT prompt for input as this will appear in your output thus causing a correct submission to be judged wrong.
- Write all output to the screen, i.e. ***stdout***, ***System.out***, ***cout*** or equivalent. Do not write to ***stderr***. Do NOT use, or even include, any module that allows direct manipulation of the screen, such as ***conio***, ***Crt***, or anything similar.
- Output from your submitted program is redirected to a file for later checking. Use of direct I/O means that such output is not redirected and hence cannot be checked thus causing a correct submission to be rejected.
- Read the questions carefully, there are several style of programming included in the problems, carefully analyze the style as using the wrong programming style will result in an incorrect submission.

# **A – Currency Converter**

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## **General Statement**

Convert a given US Dollar amount to Pounds, Lira, Francs, Marks, or Yen.

## **Input Format**

There are several test cases, the first line represent the number (n) of test cases.

The next n line of test cases consists of an integer (x) US Dollar amount, followed by the name of the currency to be converted to. ( $1 < x < 500$ )

## **Output Format**

For each case, output a dollar sign (\$), the US Dollar amount, and the words CONVERTS TO and then the converted amount followed by the currency unit name, all in uppercase letter.

## **Conversion Factor**

1USD : 0.63 POUNDS : 1404.03 LIRA : 4.76 FRANCS : 1.41 MARKS : 76.76 YEN

## **Sample Input**

```
2
17 POUNDS
228 YEN
```

## **Sample Output**

```
$17 CONVERTS TO 10.71 POUNDS
$228 CONVERTS TO 17501.28 YEN
```

# **B – Turbo Sort**

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## **General Statement**

Given the list of numbers, you are to sort them in non decreasing order.

## **Input Format**

N = the number of numbers in the list, followed by N lines of integer (X) (one in each line).

$N \leq 10^6$ ,  $0 \leq X \leq 10^6$

## **Output Format**

Output the given numbers in non decreasing order. One number per line.

## **Sample Input**

```
5
5
3
6
7
1
```

## **Sample Output**

```
1
3
5
6
7
```

## C - Big Numeric Addition

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## General Statement

Nowadays, generic calculators that existed in the market can only count numbers up to 12 digits.

However, to make calculations in the future much easier, you are requested to make one that can add two numbers with 1000 digits in length.

### Input Format

First line represents the first number.

Second line represents the second number.

## Output Format

Output the sum of the two numbers

### Sample Input

1111111111111111111111111111111111111111111111111

2222222222222222222222222222222222222222222

## Sample Output

3333333333333333333333333333333333333333333

# **D – Prime Number**

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## **General Statement**

Back in the times of war, messages needed to be encoded so that the enemy could not read them. Encryption was used and it was found that prime numbers make a better key than any old numbers.

## **Input Format**

There are several test cases, the first line represent the number (N) of test cases.

Then N test cases follow. Each test case consist of 2 integers A, B in one single line.

( $0 \leq A \leq B \leq 10000$ )

## **Output Format**

For each test cases, output every prime numbers in the range [a,b].

## **Sample Input**

```
2
1 10
10 20
```

## **Sample Output**

```
2 3 5 7
11 13 17 19
```

# **E – Even Sum**

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## **General Statement**

Scientists are trying to prove why the sums of even numbers are always even. Create a program to help them find the pattern to solve the problem.

## **Input Format**

There are several test cases, the first line represent the number (N) of test cases.

Then N test cases follow. Each test case consist of 2 integers A, B in one single line.

( $0 \leq A \leq B \leq 10000$ )

## **Output Format**

For each test cases, output the summation of all even number in the range [A,B].

## **Sample Input**

```
2
1 5
3 10
```

## **Sample Output**

```
Case 1: 6
Case 2: 18
```

# **F – Draught**

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## **General Statement**

The town BINARY has a water tank that contains 10,000 gallons of water. If there's no rain, calculate the number of weeks the water will last on weekly water usage.

## **Input Format**

The data set is on a single line. There are unknown numbers of integers in the data set. Integer 0 is used to indicate the end of data.

## **Output Format**

Use all uppercase letters. Follow the format given in the sample below.

## **Assumptions**

The weekly usage does not exceed 10,000 gallons.

The 0 used to indicate end of data is not part of the problem.

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
```



# **G – Factorial**

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## **General Statement**

Factorial function is useful in computing the number of combinations and permutations that can be constructed from a set of objects. Devise a program to output the factorial result of an integer X.

## **Input Format**

There are several test cases, the first line represent the number (N) of test cases.

Then N test cases follow. Each test case consists of 1 integer X in one single line.

( $0 \leq X \leq 21$ )

## **Output Format**

For each test case, output the factorial of X (X!).

## **Sample Input**

```
3
3
4
5
```

## **Sample Output**

```
6
24
120
```

# **H – Anything you can do, I can do better**

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## **General Statement**

You have a colleague that is extremely competitive and always tries to exceed you in any of your stories. If you say your car is fast, he will say his is faster. If you say your car is faster, he will say his is the fastest. After a few of such conversation, you realize that you can predict what he will say. Write a program to demonstrate this. Given any adjective, return its comparative form by appending “er” to it. Note that if it already ends in “e”, just append “r” to it. If it is already in its comparative form, return its superlative form, by replacing the “er” to “est”.

## **Input Format**

There are several test cases, the first line represent the number (N) of test cases.

Then N test cases follow. Each test case consists of 1 string (S) in a single line.

It will be given in all uppercase letters.

## **Output Format**

For each test case, output its comparative or superlative form. All in uppercase letters.

## **Sample Input**

```
3
COLD
BIGGER
RARE
```

## **Sample Output**

```
COLDER
BIGGEST
RARER
```

# **I – Deal!**

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## **General Statement**

BITMAP committees are having a round table meeting. X-committee members are attending the meeting. For the meeting to finish, everybody have to agree to one another. Each committee has to shake hands to one another.

Given the number X, determine the total number of handshakes that happened.

## **Input Format**

There are several test cases, the first line represent the number (N) of test cases.

Then N test cases follow. Each test case consists of 1 integer (X) in one single line.

## **Output Format**

For each test case, output total number of handshakes that occurred.

## **Sample Input**

```
2
10
12
```

## **Sample Output**

```
45
66
```

# J – Parentheses

Run-time Limit : 1 second / 5 test case

Memory Limit : 1 MB (Megabyte)

## General Statement

Mathematics study lecturer in BINUS International has been struggling to check student's work. He had to check the work of 2000 students a week. To help his work, he assigns you to create a program to check the correct usage of parentheses. Accept the string if the parenthesis closes at the right places. Reject it if there are incorrect usages.

## Input Format

There are several test cases, the first line represent the number (N) of test cases.

Then N test cases follow. Each test case consists of 1 equation; it can have integer or any string characters (including space).

## Output Format

For each test cases, output MATCH or NO MATCH depending on the usage of the parentheses whether it is correct or not.

## Sample Input

```
3
(3 + (7 * 2) – 6)
Welcome to (BITMAP 2011)
(Read the Questions (Carefully)
```

## Sample Output

```
MATCH
MATCH
NO MATCH
```