

1. Shortest and Longest Pair (MyMath) (Easy)

In this problem, you need to find the **shortest** and **longest** distance between two three-dimensional points.

Requirement:

- 1) Input is a string which contains **N** sets of numbers with “double” data type ($2 < N < 10000$). Each set contains three values: **x**, **y** and **z**.
- 2) Output number should contain **two digits** of precision of floating-point number.

Hint

You may need to include “**iomanip**” and “**sstream**” libraries in this problem.

Sample Input

1.00 1.00 1.00 1.05 1.00 1.00 3.05 4.87 5.78 3.87 4.12 8.24 7.19 100 9.57

Sample Output

0.05 99.56

// shortest distance, longest distance

2. Minesweeper(踩地雷) (MyOther) (Easy)

The goal of the Minesweeper is to find where are all the mines within a $M \times N$ ($0 < M, N \leq 1000$) field. Your goal is to **find the original map** that shows a number in a square which tells you how many mines there are adjacent to that square.

Example

Suppose the following 4×4 field with 2 mines (which are represented by an '*' character):

```
*...
....
.*..
....
```

If we would represent the same field placing the hint numbers described above, we would end up with:

```
*100
2210
1*10
1110
```

As you may have already noticed, each square may have at most 8 adjacent squares.

Hint

You may need to use "stringstream" to achieve the function of cin and cout.

Sample Input

```
4 4 *... .. .*...
3 5 ** ... .. .*...
```

Sample Output

```
*100 2210 1*10 1110
**100 33200 1*100
```

3. Get Signature (MyOther) (Normal)

You have a blank paper. There are many celebrities in the plaza. You can **only ask one celebrity** to sign on the paper. Then the celebrity each pick **one other** person they know to sign for you. (exactly one, no less, no more and never themselves). Which celebrity should you ask to **maximize** the number of signatures that you get?

Requirement:

- 1) Each case starts with a line containing an integer **N** ($2 \leq N \leq 50000$) denoting **the number of celebrities in the plaza**. Each of the next N lines contains two integers: **u v** ($1 \leq u, v \leq N, u \neq v$) meaning that celebrity **u** asks celebrity **v** to sign for you.
- 2) **Print out the celebrity m** that you should ask first to get maximize the number of signatures. If there is **more than one correct answer**, output the **smallest** number.

Sample Input

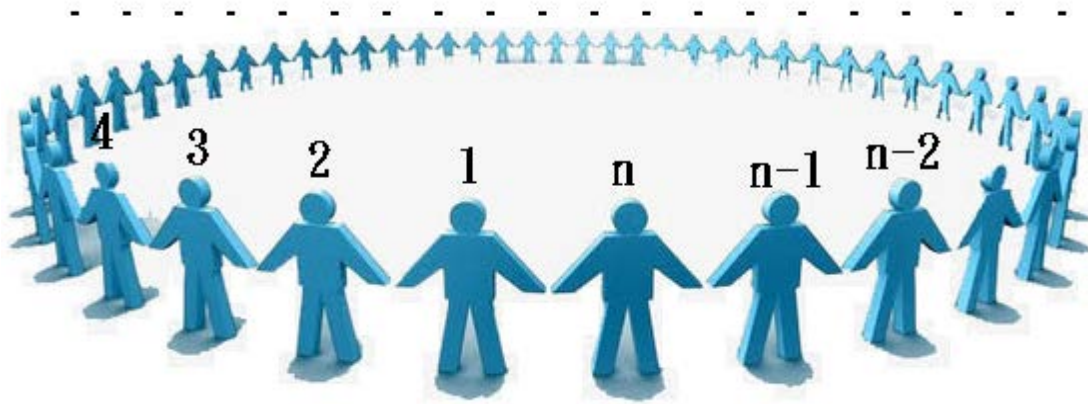
```
3 1 2 2 3 3 1
4 1 2 2 1 4 3 3 2
```

Sample Output

```
1
4
```

4. Shy Game (MyMath) (Easy)

A circle is enclosed by n people. Now we start to count m people begin from person 1, then the person who was counted at m runs away shyly. Restart the counting begin from $m+1$ again. The process will end up with only one person being there, what is his/her number?



Example

The last number of set of m . ($n = 4$, $m = 10$)

Round1: 1 2 3 4 -> starts counting from 1-> count to 10 -> 2 runs away

Round2: 1 3 4 -> starts counting from 3 -> count to 10 -> 3 runs away

Round3: 1 4 -> starts counting from 4 -> count to 10 -> 1 runs away

Answer: person 4.

Hint

You can use “queue” in this problem.

Sample Input

4 6 7 8 9 10

//The first input number represent that there are n people enclosing the circle.

//The rest of numbers represent a set of m . (Input data won't exceed 10000)

Sample Output

3 2 3 3 4

5. Fibonacci of Fibonacci (MyMath) (Easy)

Fibonacci Sequence is a function that

- $\text{Fib}(0) = 0$
- $\text{Fib}(1) = 1$
- $\text{Fib}(2) = 1$
- $\text{Fib}(n) = \text{Fib}(n - 1) + \text{Fib}(n - 2), n \geq 2$

Mr. Fib is good at calculating the ones digit(個位數) of $\text{Fib}(x)$ for any x , but he is curious of the **ones digit of $\text{Fib}(\text{Fib}(x))$** , please help him.

Example

Fibonacci series: 0 1 1 2 3 5 8 13 21

$$\text{Fib}(\text{Fib}(5))\%10 = \text{Fib}(5)\%10 = 5\%10 = 5$$

$$\text{Fib}(\text{Fib}(6))\%10 = \text{Fib}(8)\%10 = 21\%10 = 1$$

Hint

$$(a + b)\%m = ((a\%m) + (b\%m))\%m$$

Find the regular pattern of fibonacci series.

Sample Input

2 3 4 5 6

// Input are a set of x , which won't exceed INT_MAX

Sample Output

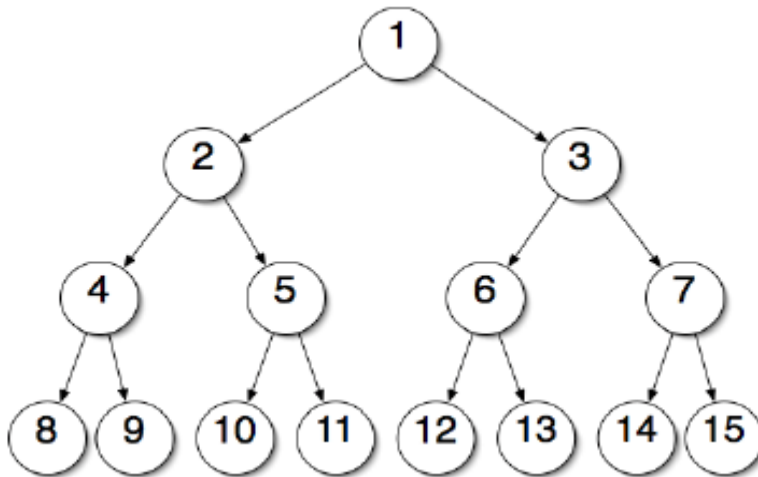
1 1 2 5 1

//output are a set of **$\text{Fib}(\text{Fib}(x))\%10$**

6. Largest root (MyTree) (Easy)

According to the diagram below, find the largest value among these subtrees.(圖中數字代表節點的編號)

Definition of subtree: find an arbitrary node, this node can construct a tree with all of its child nodes.



Each node contains **two values**, one is its **own node value**, and another is the **value of the subtree** with this node being the root.

The value of each subtree is calculated by **adding up the node value of its root and half of the sum** of the values of its two child subtree.

You need to find out which subtree has the maximum value and what is its value.

Sample input

10 20 30 40 50 60 70 80 90 100 110 120 130 140 150

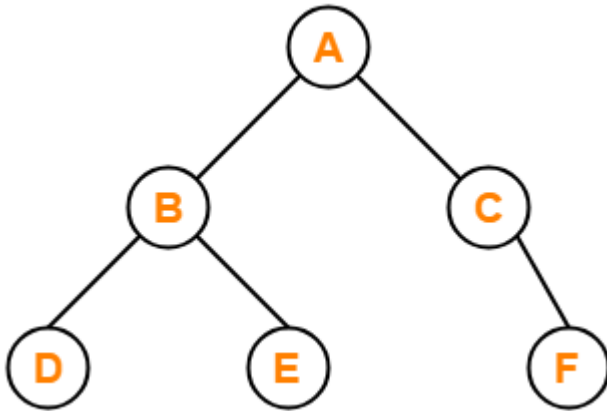
//15 values will be inputted, and the **index of each input value** correspond to the **number of nodes**, and these values aren't necessary to be positive

Sample output:

3:230.000

// display output value as a normal number(in comparison with scientific notation), and show three digits after decimal point

7. Find structures of trees (MyTree) (Hard)



There are three ways to describe trees:

- Pre-order : visit root -> go left -> go right (ABDECF)
- In-order: go left -> visit root -> go right (DBEACF)
- Post-order: go left -> go right -> visit root (DEBFCA)

Your job is finding the tree's structure according to the description of it.

Hint

You can use the structure below in your "MyTree" class

```
struct node {  
    struct node *left; //left child node  
    struct node *right; //right child node  
    string keyStr;  
    int keyInt;  
}
```

Sample input (characters of inputs will be a-z or A-Z)

ABDECF DBEACF

// input are Pre-order description first, then In-order description

Sample output

DEBFCA

// output the Post-order description of the tree that construct from input.

8. Find Factorial (MyMath) (Easy)

Factorial numbers can be express as the multiplication of zero or prime factors. Take 4!, for example. We can find that $4! = 2 \times 2 \times 2 \times 3$ and total number of prime factors is 4.

Requirement:

Given $N(N \leq 10^7 + 1)$, the number of prime factors in $X!$ (Factorial of X), you have to find the minimum possible value of X. If there is no solution, please print out "N".

Sample input:

```
1 2 4 5 7 8 9 10 11
271 279 116 253 346 352 334 333
```

Sample output:

```
2! 3! 4! 5! 6! 7! N N 8!
113! 116! N 106! 140! 143! N 135!
```


9. Shortest Summation Distance (MyMath) (Hard)

There are $2*N$ houses, you need to divide them into N groups, and each group has two houses. Let x_1 be the distance between two houses of group 1, x_2 be the distance between two houses of group 2, and so on. You have to make sure the summation $(x_1+x_2+x_3+. . .+x_N)$ is minimized.

Requirement:

- 1) Each case starts with an integer N . (N groups)
- 2) The next $2 * N$ lines will give the coordinate of the house (int x, int y).
- 3) Print out the shortest summation distance.
- 4) Output number should contain two digits of precision of floating-point number.

Sample input:

5 10 10 20 10 5 5 1 1 120 3 6 6 50 60 3 24 6 9 0 0 1 9 9 10 10
↑(N) ↑(N)

Sample output:

118.40 1.41

10. Add 1 (MyMath) (Easy)

In this problem, you need to add 1 to all elements. The input integer number lies between INT_MIN and INT_MAX-1. ($\text{INT_MIN} \leq N \leq \text{INT_MAX}-1$)

Sample Input

1 2 3 4 5 6 7 8 9

Sample Output

2 3 4 5 6 7 8 9 10

11.Distance Between Prime (MyMath) (Easy)

Given a positive integer k ($k \leq 1299709$), k lies between two prime numbers p and $p+n$, i.e., $p \leq k \leq p+n$. Your goal is to find the distance between two prime numbers p and $p+n$.

Example

Take $k=10$, for example. k lies between two prime numbers 7 and 11, i.e., $7 \leq 10 \leq 11$, so the answer is 4.

Sample Input

10 11 27 2

Sample Output

4 0 6 0

12. Mario Ojisan (MyMath) (Normal)

Mario likes coins. He travels to a country where there are **N** different types of coins. He wants to collect **as many different types of coins as** he can. Mario can withdraw **any** amount of money from the bank. He should maximize the number of different coins in a single withdrawal.

Requirement:

If we want to withdraw **X** amount of money from the bank, the bank will use the following algorithm.

```
withdraw(X){  
    if( X == 0) return;  
        //Let Y be the highest valued coin that does not exceed X.  
        //Give the customer Y valued coin.  
    withdraw(X-Y);  
}
```

- 1) Each test case starts with **N** different types of coin.
- 2) Following **N** integers represent value of each coin type.
- 3) Find the **maximum** number of different coins **in a single withdraw**.
- 4) Each test case **may not** have a coin type that the **value is 1**.

Sample input:

```
6 1 2 4 8 16 32 6 1 3 6 8 15 20  
↑(N)           ↑(N)
```

Sample output:

```
6 4
```

13.Escape! If You Want to Survive (MyOther) (hard)

Your name is Joe a janitor in the maze. One day, your ex set the fire in the maze, RUN JOE RUN for your life.

Given Joe's location in the maze and which squares of the maze are on fire, you must determine whether you can exit the maze before the fire reaches you, and find the fastest way you can escape.

Joe and the fire each move one square per minute, vertically or horizontally (not diagonally). The fire spreads all four directions from each square that is on fire. Joe may exit the maze from any square that borders the edge of the maze. Neither Joe nor the fire may enter a square that is occupied by a wall.

Requirement:

- 5) The first two integers are R and C, with $1 \leq R, C \leq 1000$.
- 6) The following R string, separated by space, of the test case each contain one row of the maze. Each of these strings contains exactly C characters.
- 7) #=walls, .=place can pass, J=Joe, F=place on fire.
- 8) Output the fastest time Joe can safely exit the maze in minutes.
- 9) If there is not possible way to escape, please print "N".

Example

```
4 4
####      ####      ####
#JF#      #FF#      #FF#
#..# =>   #JF# =>   #FF# =>   Escaped!   =>3 min
#..#      #..#      #JF#
```

Sample input:

```
4 4 ##### #JF# #..# #..#
```

Sample output:

```
3
```

14.Subtract 1 (MyMath) (Easy)

In this problem, you need to subtract 1 to all elements. The input integer number lies between $\text{INT_MIN}+1$ and INT_MAX . ($\text{INT_MIN}+1 \leq N \leq \text{INT_MAX}$)

Sample Input

1 2 3 4 5 6 7 8 9 10

Sample Output

0 1 2 3 4 5 6 7 8 9

15.Find the Period of String (MyString) (Easy)

When a string repeats every N characters, it can be said that the string has a period N. In this problem, you need to find the **shortest** period of the string.

Example

Take the string "abcabcabcabc", for example. The string has repeated strings "abc", "abcabc", "abcabcabc", and the corresponding period is 3,6,12. We want to find the shortest period, so the answer is 3.

Sample Input

abbabb

abcdef

ababa

Sample Output

3

6

5

16.Large factorial (MyMath) (Easy)

Your job is to calculate the value of $n!$.

Sample input: $(0 \leq n < 10000)$

20

Sample output

2432902008176640000

17.Sum of large hexadecimal integers (MyMath) (Normal)

Calculate the sum of large hexadecimal integers

Sample input: (only lower-case character will be inputted)

ffffffff 10 1 f 0

//arbitrary number of data will be inputted, terminated by a single zero

Sample output

100000001f

//please use lower-case character in your output result

18.Longest Common Substring (MyString) (Easy)

Given two strings, you should capitalize all the character and remove all the punctuations including **comma(,)**, **period(.)**, **question mark(?)**, and **exclamation mark(!)**, then find the **longest common substring** between two strings. If there are more than one longest common substring between two strings, please print the first of them in **dictionary order**.

Sample Input

abcdefgh123456abcde vwxyz123456vwxyz

a,bc!defg?h12..345,.?6abc!!de vwx,.?yz12,3.45!!?6v.wx,!?yz!

Sample Output

123456

// Longest common substring

123456

// Longest common substring

19. Get Solution (MyMath) (Normal)

Given a $N \times N$ ($1 \leq N \leq 10$) square matrix **A** and a $N \times 1$ column vector **b**. Please solve the equation $\mathbf{Ax} = \mathbf{b}$ by Gaussian elimination and output the solution **x**. You don't need to consider singularity in Gaussian elimination but you should take **row exchange** into consideration.

In this problem, all elements in **A** and **b** are floating numbers. The output format contains **two digits of precision** of floating-point numbers.

Sample Input

```
3 2 4 -2 4 9 -3 -2 -3 7 2 8 10
3 2 1 0 1 2 1 0 1 2 30.35 32.67 23.65
//N, first row, second row, ..., Nth row, b
```

Sample Output

```
-1.00 2.00 2.00
12.34 5.67 8.99
```

20.Longest Peek Palindrome (MyString) (Easy)

Given two strings, you should capitalize all the characters, remove all the punctuations including **comma(,)**, **period(.)**, **question mark(?)**, and **exclamation mark(!)**, then find the **longest peek palindrome**. If there are more than one longest peek palindrome in one string, please print the first of them in dictionary order.

Example

if the input string is "abcb", the peek palindrome contains "a", "b", "c", "bcb". You should print "bcb" on the screen.

Sample Input

```
TO,T
A!?b,cd,CB.A
2.A3,M,EA.S
A,BC,D.E!?FG
```

Sample Output

```
TOT
ABCD CBA
2
A
```

21.Maze (MyOthers) (Normal)

In this problem, we give you a maze which includes a starting point and an end point. You need to follow the **specify way** to find the path from the starting point to the end point, then print out the path you find.

- 1) The given maze must have a path from the starting point to the end point, and the outer ring of the maze is the wall.
 - a) 2: It is the wall and you can't walk through it.
 - b) 0: The place you can walk
 - c) S: Starting point
 - d) E: End point
- 2) Moving rules:
 - a) For a path, the position you walked through **cannot be repeated**.
 - b) Start from the beginning.
 - c) Find the next legal move in the order of **right, bottom, left and top**.
 - d) Repeat **Step C** until all directions are not allowed to walk. Go back to last step, then repeat **Step C** until you find the end point.
- 3) Output format: Use "1" to indicate the path you find.

Hint

- 1) You can use **Stack** to solve this problem
- 2) The idea for this problem:
 - a) When you find a legal move, **save the current position** into a **stack<Point>** in order to avoid running into a dead end.
 - b) Use an **array mark[][]** to **record** whether you've visited the position or not to avoid tracing the same path twice
 - c) When running into a dead end, use **Stack** to go back to last step.
 - d) After reaching the end point, use **Stack** to show the path.
- 3) You may need a Class Point to represent your position
- 4) <http://www.cplusplus.com/reference/stack/stack/>

Example:

7 // each row has 7 elements
6 // 6 rows

2 2 2 2 2 2 2
2 0 S 0 0 0 2
2 0 0 0 0 0 2
2 0 0 0 0 2 2
2 0 2 2 0 E 2
2 2 2 2 2 2 2

ANS:

2 2 2 2 2 2 2
2 0 S 1 1 1 2
2 0 0 0 1 1 2
2 0 0 0 1 2 2
2 0 2 2 1 E 2
2 2 2 2 2 2 2

Sample Input

7 6 2 2 2 2 2 2 2 2 0 S 0 0 0 2 2 0 0 0 0 0 2 2 0 0 0 0 2 2 2 0 2 2 0 E 2 2 2 2 2 2
2 2 (No line breaks “/n”)

Sample Output

2 2 2 2 2 2 2 2 0 S 1 1 1 2 2 0 0 0 1 1 2 2 0 0 0 1 2 2 2 0 2 2 1 E 2 2 2 2 2 2 2 2

22.The Easy City 1 (MyDatabase) (Easy)

Please implement the C++ code using SQL command in QT, and follow the steps to output the final result: (You have to establish a connection with MySQL Server first.)

- 1) Drop the schema "CITYDATABASE" if it exists in MySQL server.
- 2) Create a new schema "CITYDATABASE" in MySQL server.
- 3) Set "CITYDATABASE" as default database.
- 4) Drop the table "CITYTABLE" if it exists in "CITYDATABASE".
- 5) Create a new table "CITYTABLE" described as follows:

| Field | Type |
|---------|----------------------|
| ID | Number (Primary Key) |
| COUNTRY | VARCHAR (50) |
| CITY | VARCHAR (60) |
| LAT | NUMBER |
| LON | NUMBER |

Note: LAT is the latitude and LON is the longitude.

- 6) Load data from "city.csv" into table "CITYTABLE". The data format in "city.csv" is as following example:

ID,Country,City,Latitude,Longitude

100,"Taipei","Zhongzheng",121.5198839,25.03240487

130,"Taipei","Datong",121.5130417,25.06342433

104,"Taipei","Zhongshan",121.5381597,25.06969917

150,"Taipei","Songshan",121.5575876,25.05999101

106,"Taipei","Da'an",121.5434446,25.02677012

108,"Taipei","Wanhua",121.4979858,25.02858990

110,"Taipei","Xinyi",121.5716697,25.03062083

111,"Taipei","Shilin",121.5508473,25.12546704

112,"Taipei","Beitou",121.5177992,25.14806820

114,"Taipei","Neihu",121.5923828,25.08370623

115,"Taipei","Nangang",121.6097573,25.03600934

116,"Taipei","Wenshan",121.5736082,24.98857934

200,"Keelung","Ren'ai",121.7434205,25.11945421

201,"Keelung","Xinyi",121.772646,25.125765790

202,"Keelung","Zhongzheng",121.7783549,25.14365754

203,"Keelung","Zhongshan",121.7308913,25.14986365

204,"Keelung","Anle",121.7078325,25.14139521
205,"Keelung","Nuannuan",121.7447344,25.08097003
206,"Keelung","Qidu",121.683628,25.109620280
556,"Nantou","Xinyi",121.0212867,23.6554647

Please use only one query statement to solve each requirement.

- 7) Query the difference (**dcc**) between the total number of city and the number of distinct city with the specified data requirements:
 - a) The names of city start as specified characters (**ch**) and end with no vowels (i.e., a, e, i, o, and u). The specified starting character can be one of multiple characters in a specified range.
 - b) The latitude is greater than **la** and the longitude is smaller than **lo**.
- 8) Update the values of latitudes to **p** times and the values of longitude to **q** times if the id number is even/odd (**ev/od**).
- 9) Query the sum of the values in latitude (**sla**), the sum of the values in longitude (**slo**), and the difference between maximum length of city name and minimum length of city name (**dmm**). The query data limitations are as follows:
 - a) The first **m even/odd** ID numbers with ascending/descending (**asc/des**) length of city name and reverse alphabetical order.
 - b) Both summations of latitude and longitude must be rounded to a scale of **n (n<=5)** decimal places.

Input Format

ch la lo p q ev/od m even/odd asc/des n

Output Format

dcc sla slo dmm

Sample Input

nz 121.5 25.3 2 5 od 5 even des 3

Sample Output

2 607.968 125.294 3

23. Manhattan Distance (MyDatabase) (Medium)

Please implement the C++ code using SQL command in QT, and follow the steps to output the final result: (You have to establish a connection with MySQL Server first.)

- 1) Drop the schema "CITYDATABASE" if it exists in MySQL server.
- 2) Create a new schema "CITYDATABASE" in MySQL server.
- 3) Set "CITYDATABASE" as default database.
- 4) Drop the table "CITYTABLE" if it exists in "CITYDATABASE".
- 5) Create a new table "CITYTABLE" described as follows:

| Field | Type |
|---------|----------------------|
| ID | Number (Primary Key) |
| COUNTRY | VARCHAR (50) |
| CITY | VARCHAR (60) |
| LAT | NUMBER |
| LON | NUMBER |

Note: LAT is the latitude and LON is the longitude.

- 6) Load data from "city.csv" into table "CITYTABLE". The data format in "city.csv" is as follows:

ID,Country,City,Latitude,Longitude

100,"Taipei","Zhongzheng",121.5198839,25.03240487

130,"Taipei","Datong",121.5130417,25.06342433

104,"Taipei","Zhongshan",121.5381597,25.06969917

150,"Taipei","Songshan",121.5575876,25.05999101

106,"Taipei","Da'an",121.5434446,25.02677012

108,"Taipei","Wanhua",121.4979858,25.02858990

110,"Taipei","Xinyi",121.5716697,25.03062083

111,"Taipei","Shilin",121.5508473,25.12546704

112,"Taipei","Beitou",121.5177992,25.14806820

114,"Taipei","Neihu",121.5923828,25.08370623

115,"Taipei","Nangang",121.6097573,25.03600934

116,"Taipei","Wenshan",121.5736082,24.98857934

200,"Keelung","Ren'ai",121.7434205,25.11945421

201,"Keelung","Xinyi",121.772646,25.125765790

202,"Keelung","Zhongzheng",121.7783549,25.14365754

203,"Keelung","Zhongshan",121.7308913,25.14986365

204,"Keelung","Anle",121.7078325,25.14139521
205,"Keelung","Nuannuan",121.7447344,25.08097003
206,"Keelung","Qidu",121.683628,25.109620280
556,"Nantou","Xinyi",121.0212867,23.6554647

Please use only one query statement to solve the requirement.

7) First, find the data in the range of the s^{th} to the t^{th} largest latitudes which is less than $v1$, and then define $x1$ and $x2$ be the smallest and the largest longitudes in that range. Second, find the data in the range of the s^{th} to the t^{th} smallest longitudes which is larger than $v2$, and then define $y1$ and $y2$ be the smallest and the largest latitudes in that range. Please query the Manhattan Distance (md) between points $(x1, y1)$ and $(x2, y2)$. The answer should be rounded to a scale of 4 decimal places. The formula of Manhattan Distance is $|x1 - x2| + |y1 - y2|$. You can use `ABS()` function in MySQL.

Input Format

s t $v1$ $v2$

Output Format

md

Sample Input

4 9 121.7 25

Sample Output

0.3686

24.The Median (MyDatabase) (Hard)

Please implement the C++ code using SQL command in QT, and follow the steps to output the final result: (You have to establish a connection with MySQL Server first.)

- 1) Drop the schema "CITYDATABASE" if it exists in MySQL server.
- 2) Create a new schema "CITYDATABASE" in MySQL server.
- 3) Set "CITYDATABASE" as default database.
- 4) Drop the table "CITYTABLE" if it exists in "CITYDATABASE".
- 5) Create a new table "CITYTABLE" described as follows:

| Field | Type |
|---------|----------------------|
| ID | Number (Primary Key) |
| COUNTRY | VARCHAR (50) |
| CITY | VARCHAR (60) |
| LAT | NUMBER |
| LON | NUMBER |

Note: LAT is the latitude and LON is the longitude.

- 6) Load data from "city.csv" into table "CITYTABLE". The data format in "city.csv" is as follows:

ID,Country,City,Latitude,Longitude

100,"Taipei","Zhongzheng",121.5198839,25.03240487

130,"Taipei","Datong",121.5130417,25.06342433

104,"Taipei","Zhongshan",121.5381597,25.06969917

150,"Taipei","Songshan",121.5575876,25.05999101

106,"Taipei","Da'an",121.5434446,25.02677012

108,"Taipei","Wanhua",121.4979858,25.02858990

110,"Taipei","Xinyi",121.5716697,25.03062083

111,"Taipei","Shilin",121.5508473,25.12546704

112,"Taipei","Beitou",121.5177992,25.14806820

114,"Taipei","Neihu",121.5923828,25.08370623

115,"Taipei","Nangang",121.6097573,25.03600934

116,"Taipei","Wenshan",121.5736082,24.98857934

200,"Keelung","Ren'ai",121.7434205,25.11945421

201,"Keelung","Xinyi",121.772646,25.125765790

202,"Keelung","Zhongzheng",121.7783549,25.14365754

203,"Keelung","Zhongshan",121.7308913,25.14986365

204,"Keelung","Anle",121.7078325,25.14139521
205,"Keelung","Nuannuan",121.7447344,25.08097003
206,"Keelung","Qidu",121.683628,25.109620280
556,"Nantou","Xinyi",121.0212867,23.6554647

Please use only one query statement to solve the requirement.

- 7) The median is the value separating the larger half from the smaller half of a data sample. Please query the median (**med**) of the range of the **ath** to the **bth** latitudes/longitudes (**lat/lon**) with the following dataset: the last number of id of the corresponding cities is **m**, and they are ordered by latitudes/longitudes (**lat/lon**) in ascending/descending (**asc/des**) order. The answer should be rounded to a scale of 4 decimal places.

Input Format

a b lat/lon m asc/des

Output Format

med

Sample Input

1 4 lon 6 des

Sample Output

25.0077

25.The Easy City 2 (MyDatabase) (Easy)

Please implement the C++ code using SQL command in QT, and follow the steps to output the final result: (You have to establish a connection with MySQL Server first.)

- 10) Drop the schema "CITYDATABASE" if it exists in MySQL server.
- 11) Create a new schema "CITYDATABASE" in MySQL server.
- 12) Set "CITYDATABASE" as default database.
- 13) Drop the table "CITYTABLE" if it exists in "CITYDATABASE".
- 14) Create a new table "CITYTABLE" described as follows:

| Field | Type |
|---------|----------------------|
| ID | Number (Primary Key) |
| COUNTRY | VARCHAR (50) |
| CITY | VARCHAR (60) |
| LAT | NUMBER |
| LON | NUMBER |

Note: LAT is the latitude and LON is the longitude.

- 15) Load data from "city.csv" into table "CITYTABLE". The data format in "city.csv" is as following example:

ID,Country,City,Latitude,Longitude

100,"Taipei","Zhongzheng",121.5198839,25.03240487

130,"Taipei","Datong",121.5130417,25.06342433

104,"Taipei","Zhongshan",121.5381597,25.06969917

150,"Taipei","Songshan",121.5575876,25.05999101

106,"Taipei","Da'an",121.5434446,25.02677012

108,"Taipei","Wanhua",121.4979858,25.02858990

110,"Taipei","Xinyi",121.5716697,25.03062083

111,"Taipei","Shilin",121.5508473,25.12546704

112,"Taipei","Beitou",121.5177992,25.14806820

114,"Taipei","Neihu",121.5923828,25.08370623

115,"Taipei","Nangang",121.6097573,25.03600934

116,"Taipei","Wenshan",121.5736082,24.98857934

200,"Keelung","Ren'ai",121.7434205,25.11945421

201,"Keelung","Xinyi",121.772646,25.125765790

202,"Keelung","Zhongzheng",121.7783549,25.14365754

203,"Keelung","Zhongshan",121.7308913,25.14986365

204,"Keelung","Anle",121.7078325,25.14139521
205,"Keelung","Nuannuan",121.7447344,25.08097003
206,"Keelung","Qidu",121.683628,25.109620280
556,"Nantou","Xinyi",121.0212867,23.6554647

Please use only one query statement to solve each requirement.

- 16) The size of a country is defined by the number of cities. Query the first three characters of the n^{th} largest/smallest (**lar/sma**) country name (**sn**), and also query the number of the cities (**nc**) which that country has.
- 17) Delete the data which meet one of following requirements:
 - a) The names of city end with specified characters (**che**) or do not start with specified characters (**chs**). The specified character can be one of multiple characters in a specified range.
 - b) The latitude is in the range [**la**, **ha**] and the longitude is in the range [**lo**, **ho**].
- 18) Swap the value of latitude and the value of longitude if the last number of id is **m**.
- 19) If x_1 and x_2 are the largest value and the smallest value of longitudes, and y_1 and y_2 are the largest value and the smallest value of latitudes, please query the Euclidean Distance (**ed**) between points (x_1, y_1) and (x_2, y_2). The answer should be rounded to a scale of 4 decimal places. You can use POW(), SQRT() functions in MySQL.

Input Format

n lar/sma chs che la ha lo ho m

Output Format

sn nc ed

Sample Input

2 sma nz aeiou 121.5 131.5 25.3 30.3 5

Sample Output

Kee 7 136.7931

26.Longest/Shortest Distance (MyDatabase) (Medium)

Please implement the C++ code using SQL command in QT, and follow the steps to output the final result: (You have to establish a connection with MySQL Server first.)

- 8) Drop the schema "CITYDATABASE" if it exists in MySQL server.
- 9) Create a new schema "CITYDATABASE" in MySQL server.
- 10) Set "CITYDATABASE" as default database.
- 11) Drop the table "CITYTABLE" if it exists in "CITYDATABASE".
- 12) Create a new table "CITYTABLE" described as follows:

| Field | Type |
|---------|----------------------|
| ID | Number (Primary Key) |
| COUNTRY | VARCHAR (50) |
| CITY | VARCHAR (60) |
| LAT | NUMBER |
| LON | NUMBER |

Note: LAT is the latitude and LON is the longitude.

- 13) Load data from "city.csv" into table "CITYTABLE". The data format in "city.csv" is as follows:

ID,Country,City,Latitude,Longitude

100,"Taipei","Zhongzheng",121.5198839,25.03240487

130,"Taipei","Datong",121.5130417,25.06342433

104,"Taipei","Zhongshan",121.5381597,25.06969917

150,"Taipei","Songshan",121.5575876,25.05999101

106,"Taipei","Da'an",121.5434446,25.02677012

108,"Taipei","Wanhua",121.4979858,25.02858990

110,"Taipei","Xinyi",121.5716697,25.03062083

111,"Taipei","Shilin",121.5508473,25.12546704

112,"Taipei","Beitou",121.5177992,25.14806820

114,"Taipei","Neihu",121.5923828,25.08370623

115,"Taipei","Nangang",121.6097573,25.03600934

116,"Taipei","Wenshan",121.5736082,24.98857934

200,"Keelung","Ren'ai",121.7434205,25.11945421

201,"Keelung","Xinyi",121.772646,25.125765790

202,"Keelung","Zhongzheng",121.7783549,25.14365754

203,"Keelung","Zhongshan",121.7308913,25.14986365

204,"Keelung","Anle",121.7078325,25.14139521
205,"Keelung","Nuannuan",121.7447344,25.08097003
206,"Keelung","Qidu",121.683628,25.109620280
556,"Nantou","Xinyi",121.0212867,23.6554647

Please use only one query statement to solve the requirement.

- 14) Please query the longest distance (**ld**) and the shortest distance (**sd**) between two cities according to the following requirements:
- a) In the beginning, the latitudes and longitudes must be rounded to **nth** ($n \leq 5$) decimal place. Therefore, the duplicated positions must be truncated (the first duplicated position in the database is reserved).
 - b) Choose the data in the range of the **ath** to the **bth** cities ordered by city names in alphabetical/reverse alphabetical (**al/ra**) order and id in ascending order sequentially to calculate the longest distance and shortest distance.
 - c) The answer should be rounded to a scale of 4 decimal places.

Input Format

n a b al/ra

Output Format

ld sd

Sample Input

1 3 5 al

Sample Output

1.6125 0.2236

27.Symmetric Pairs (MyDatabase) (Hard)

Please implement the C++ code using SQL command in QT, and follow the steps to output the final result: (You have to establish a connection with MySQL Server first.)

- 8) Drop the schema "CITYDATABASE" if it exists in MySQL server.
- 9) Create a new schema "CITYDATABASE" in MySQL server.
- 10) Set "CITYDATABASE" as default database.
- 11) Drop the table "CITYTABLE" if it exists in "CITYDATABASE".
- 12) Create a new table "CITYTABLE" described as follows:

| Field | Type |
|---------|----------------------|
| ID | Number (Primary Key) |
| COUNTRY | VARCHAR (50) |
| CITY | VARCHAR (60) |
| LAT | NUMBER |
| LON | NUMBER |

Note: LAT is the latitude and LON is the longitude.

- 13) Load data from "city.csv" into table "CITYTABLE". The data format in "city.csv" is as follows:

ID,Country,City,Latitude,Longitude

100,"Taipei","Zhongzheng",121.5198839,25.03240487

130,"Taipei","Datong",121.5130417,25.06342433

104,"Taipei","Zhongshan",121.5381597,25.06969917

150,"Taipei","Songshan",121.5575876,25.05999101

106,"Taipei","Da'an",121.5434446,25.02677012

108,"Taipei","Wanhua",121.4979858,25.02858990

110,"Taipei","Xinyi",121.5716697,25.03062083

111,"Taipei","Shilin",121.5508473,25.12546704

112,"Taipei","Beitou",121.5177992,25.14806820

114,"Taipei","Neihu",121.5923828,25.08370623

115,"Taipei","Nangang",121.6097573,25.03600934

116,"Taipei","Wenshan",121.5736082,24.98857934

200,"Keelung","Ren'ai",121.7434205,25.11945421

201,"Keelung","Xinyi",121.772646,25.125765790

202,"Keelung","Zhongzheng",121.7783549,25.14365754

203,"Keelung","Zhongshan",121.7308913,25.14986365

204,"Keelung","Anle",121.7078325,25.14139521
205,"Keelung","Nuannuan",121.7447344,25.08097003
206,"Keelung","Qidu",121.683628,25.109620280
556,"Nantou","Xinyi",121.0212867,23.6554647

Please use only one query statement to solve each requirement.

- 14) Swap the value of latitude and the value of longitude if the last number of id is even/odd (ev/od) number.
- 15) Update the value of latitude to be equal to the value of longitude in CITYTABLE if the last number of id is m.
- 16) If there are two points (X1, Y1) and (X2, Y2) that $X1 = Y2$ and $X2 = Y1$, they are said to be the symmetric pair. First, make the values of latitude and the value of longitude to be rounded to the n^{th} ($n \leq 5$) decimal place. And then let the rounded values of latitude be X and the rounded values of longitude be Y, please write a query to output X (x) and Y (y) of k^{th} symmetric pair ordered by X and Y sequentially. Note that you should truncate the duplicated points, and the X of output answer is smaller than or equal to the Y of output.

For following example, the ordered symmetric pair are (10, 10), (10,20).

| X | Y |
|----|----|
| 10 | 10 |
| 10 | 10 |
| 10 | 20 |
| 20 | 10 |
| 10 | 20 |
| 15 | 20 |

Input Format

ev/od m n k

Output Format

x y

Sample Input

od 6 1 4

Sample Output

25.1 121.7