[PT] arrays

J - Top 3

Escreva um programa que leia o nome e salário dos funcionários de uma empresa. A leitura termina quando for inserido **"END"** como o nome de um funcionário.

De seguida, calcule e mostre o top três (3) dos funcionários mais bem pagos. Caso haja igualdade de vencimentos, a escolha deve seguir a ordenação alfabética crescente do nome.

Mostre em linhas separadas o nome e salário de cada funcionário, ordenados de forma decrescente do salário, no seguinte formato:

"#<ordem>:<nome>:<salario>"

<ordem> indica o lugar no top.
<nome> nome do funcionário
<salario> valor do salário

[EN] arrays

J - Top 3

Write a program to read the name and salary of a company's employees and display the top three (3) highest paid employees.

The program should start by reading, continuously, the name and salary of each employee. Reading ends when the name **"END"** is entered as the name of an employee.

Then, the top three (3) highest paid employees must be calculated and displayed. If there are employees with the equal salary, the choice should follow the ascending alphabetical order of the name.

Each employee of the top three (if any) should be displayed on separate lines. The name and salary of each employee must be displayed in descending order of salary, in the following format:

"#<top order>:<name>:<salary>

Example1:

Input	Output
Ana Anacleta 1000 Berta Barata 1000 Carla Costa 100 Luisa Lima 125 Elsa Eira 5000 Francisca Fe 123 Gloria Gomes 55 Hercilia Hora 250 END	#1:Elsa Eira:5000 #2:Ana Anacleta:1000 #3:Berta Barata:1000

Example2:

Input	Output
Hercilia Hora 1000 Elsa Eira 1000 Gloria Gomes 1000 Luisa Lima 1000 Berta Barata 1000 Francisca Fé 1000 Carla Costa 1000 Ana Anacleta 1000	#1:Ana Anacleta:1000 #2:Berta Barata:1000 #3:Carla Costa:1000

END	

Example3:

Input	Output
Hercilia Hora 100 Elsa Eira 1000 END	#1:Elsa Eira:1000 #2:Hercilia Hora:100

Example4:

Input	Output
Ana Anacleta 1000 Carla Costa 1000 Berta Barata 100 Ana Anacleta 125 Elsa Eira 5000 Francisca Fé 123 Ana Anacleta 55 Hercilia Hora 250 END	#1:Elsa Eira:5000 #2:Ana Anacleta:1000 #3:Carla Costa:1000

Example5:

Input	Output
Ana Anacleta 1000 Berta Barata 1000 Carla Costa 100 Luisa Lima 125 Elsa Eira 5000 Francisca Fé 123 Gloria Gomes 55 Hercilia Hora 250 Ana Anacleta 1000 Berta Barata 1000 Carla Costa 100 Luisa Lima 125 Elsa Eira 5000 Francisca Fé 123 Gloria Gomes 55 Hercilia Hora 250 Ana Anacleta 1000 Berta Barata 1000 Carla Costa 100 Luisa Lima 125 Elsa Eira 5000 Francisca Fé 123 Gloria Gomes 55 Hercilia Hora 250	#1:Elsa Eira:5000 #2:Elsa Eira:5000 #3:Elsa Eira:5000

Ana Anacleta 1000 Berta Barata 1000 Carla Costa 100 Luisa Lima 125 Elsa Eira 5000 Francisca Fé 123 Gloria Gomes 55 Hercilia Hora 250 END	

[PT] arrays

K - Sem repetições

Escreva um programa que leia um conjunto N de números inteiros e os visualize pela mesma ordem mas sem repetições.
O programa deve ler, primeiro, o valor de N e, de seguida, os N números do conjunto.

Cada um dos elementos do conjunto resultado deve ser visualizado em linhas separadas.

[EN] arrays

K - No repetitions

Write a program that reads N integers and displays them in the same order but without repetitions.
First, the program must read the value of N. Next, it must read the N numbers of the set.

Each element of the result set must be displayed on separate lines.

Example1:

Input	Output
7 10 20 50 10 10 60 20	10 20 50 60

Example2:

Input	Output
20 1 9 0 2 1 3 1 2 0 1 2 1 5 6 7 9 0 1 2 3	1 9 0 2 3 5 6 7

Example3:

Input	Output

20 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1

[PT] arrays

L - Vizinhos

Escreva um programa que leia um conjunto números inteiros e visualize todos os números maiores que os seus vizinhos.
Um número é maior que os seus vizinhos quando é, simultaneamente, maior que o número anterior e que o número seguinte.
A leitura termina quando for inserido um número negativo. O tamanho do conjunto de números nunca será superior a 30.

Os elementos do resultado devem ser visualizados em linhas separadas.

[EN] arrays

L - Neighbours

Write a program to read a set of integers and display all numbers in that set that are larger than their neighbours.
A number is larger than its neighbours if it is simultaneously larger than the previous and next number.
Reading ends when a negative number is entered. Never more than 30 numbers will be entered.

Result elements should be displayed on separate lines.

Example1:
8, 2, **4** , 1, 6, **12** , 5, 9, -1

Input	Output
8 2 4 1 6 12 5 9 -1	4 12

Example2:

Input	Output
1 **9** 2 3 4 5 **6** 1 1212 1212 1211 1210 1111 1111	9 6 1112 9 11

1112 10 9 8 7 **9** 7 7 7 6 **11** 1 -5	

Example3:

Input	Output
10 20 20 21 21 21 20 19 10 5 4 3 2 1 -1	

[PT] arrays

M - Algarismos diferentes

Escreva um programa que leia um conjunto números inteiros positivos e visualize, para cada um, quantos algarismos diferentes o constituem.
A leitura termina quando for inserido um número negativo.

Os elementos do resultado devem ser visualizados em linhas separadas, no seguinte formato:

<nº inserido>:<algarismos diferentes>

[EN] arrays

M - Different digits

Write a program that reads a set of positive integers and visualizes, for each one, how many different digits make up the number.
Reading ends when a negative number is entered.

Result elements must be displayed on separate lines, in the following format:

<entered number>:<different digits>

Example1:

Input	Output
12123518 2222 400 12345 -1	12123518:5 2222:1 400:2 12345:5

Example2:

Input	Output
111111111 102030405 500000005 987654321 919191919 200000000	111111111:1 102030405:6 500000005:2 987654321:9 919191919:2 200000000:2

100002000	100002000:3
778877997	778877997:3
102938475	102938475:9
0	0:1
1	1:1
2	2:1
3	3:1
4	4:1
5	5:1
6	6:1
7	7:1
8	8:1
9	9:1
10	10:2
11	11:1
12	12:2
13	13:2
14	14:2
15	15:2
16	16:2
17	17:2
18	18:2
19	19:2
20	20:2
-11	

Example3:

Input	Output
1000000	1000000:2
1001001	1001001:2
1002002	1002002:3
1003003	1003003:3
1004004	1004004:3
1005005	1005005:3
1006006	1006006:3
1007007	1007007:3
1008008	1008008:3
1009009	1009009:3
1010010	1010010:2
1011011	1011011:2
1012012	1012012:3
1013013	1013013:3
1014014	1014014:3
1015015	1015015:3
1016016	1016016:3
1017017	1017017:3
1018018	1018018:3
1019019	1019019:3
1020020	1020020:3
1021021	1021021:3
1022022	1022022:3
1023023	1023023:4
1024024	1024024:4
1025025	1025025:4
1026026	1026026:4
1027027	1027027:4
1028028	1028028:4
1029029	1029029:4
1030030	1030030:3
1031031	1031031:3
1032032	1032032:4
1033033	1033033:3
1034034	1034034:4
1035035	1035035:4
1036036	1036036:4
1037037	1037037:4
1038038	1038038:4
1039039	1039039:4
1040040	1040040:3
1041041	1041041:3
1042042	1042042:4
1043043	1043043:4
1044044	1044044:3
1045045	1045045:4
1046046	1046046:4
1047047	1047047:4
1048048	1048048:4
1049049	1049049:4
-1234	

[PT] arrays

N - Diagonais

Escreva um programa que leia uma matriz quadrada de números inteiros e escreva os valores correspondentes às várias diagonais com o mesmo sentido da diagonal principal e da direita para a esquerda.
O programa deve ler um valor N (1 < N <= 20) correspondendo à dimensão da matriz. De seguida, deve ler N linhas contendo, cada uma, N números separados por espaços.

Os elementos de cada diagonal devem ser visualizados em linhas separadas e na seguinte forma:

[a]...[z]

[EN] arrays

N - Diagonals

Write a program that reads a square matrix of integers and writes the values corresponding to the various diagonals with the same direction of the main diagonal and from right to left.
The program should read a value N (1 < N <= 20) corresponding to the dimension of the array. Then it should read N lines each containing N numbers separated by spaces.

The elements of each diagonal must be visualized in separate lines and in the following way:

[a]...[z]

Example1:

Input	Output
10 1 2 3 4 5 6 7 8 9 10 2 3 4 5 6 7 8 9 10 11 3 4 5 6 7 8 9 10 11 12 4 5 6 7 8 9 10 11 12 13 5 6 7 8 9 10 11 12 13 14 6 7 8 9 10 11 12 13 14 15 7 8 9 10 11 12 13 14 15 16 8 9 10 11 12 13 14 15 16 17 9 10 11 12 13 14 15 16 17 18 10 11 12 13 14 15 16 17 18 19	[10] [9][11] [8][10][12] [7][9][11][13] [6][8][10][12][14] [5][7][9][11][13][15] [4][6][8][10][12][14][16] [3][5][7][9][11][13][15][17] [2][4][6][8][10][12][14][16][18] [1][3][5][7][9][11][13][15][17][19] [2][4][6][8][10][12][14][16][18] [3][5][7][9][11][13][15][17] [4][6][8][10][12][14][16] [5][7][9][11][13][15] [6][8][10][12][14] [7][9][11][13] [8][10][12] [9][11] [10]

Example2:

Input	Output
3 10 20 30 50 60 70 90 100 110	[30] [20][70] [10][60][110] [50][100] [90]

Example3:

Input	Output
11 1 2 3 4 5 6 7 8 9 10 11 21 22 23 24 25 26 27 28 29 30 31 31 32 33 34 35 36 37 38 39 40 41 41 42 43 44 45 46 47 48 49 50 51 51 52 53 54 55 56 57 58 59 60 61 61 62 63 64 65 66 67 68 69 70 71 71 72 73 74 75 76 77 78 79 80 81 81 82 83 84 85 86 87 88 89 90 91 91 92 93 94 95 96 97 98 99 100 101 101 102 103 104 105 106 107 108 109 110 111 111 112 113 114 115 116 117 118 119 120 121	[11] [10][31] [9][30][41] [8][29][40][51] [7][28][39][50][61] [6][27][38][49][60][71] [5][26][37][48][59][70][81] [4][25][36][47][58][69][80][91] [3][24][35][46][57][68][79][90][101] [2][23][34][45][56][67][78][89][100][111] [1][22][33][44][55][66][77][88][99][110][121] [21][32][43][54][65][76][87][98][109][120] [31][42][53][64][75][86][97][108][119] [41][52][63][74][85][96][107][118] [51][62][73][84][95][106][117] [61][72][83][94][105][116] [71][82][93][104][115] [81][92][103][114] [91][102][113] [101][112]

	[111]

Example4:

Input	Output
20 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 4 5 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 5 6 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 6 7 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 7 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 8 9 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 9 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 4 5 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 5 6 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 6 7 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 7 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 8 9 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 9	[0] [1][1] [1][1][2] [1][1][1][3] [1][1][1][1][4] [1][1][1][1][1][5] [1][1][1][1][1][1][6] [1][1][1][1][1][1][1][7] [1][1][1][1][1][1][1][1][8] [1][1][1][1][1][1][1][1][1][9] [1][1][1][1][1][1][1][1][1][1][0] [1][1][1][1][1][1][1][1][1][1][1][1] [1][1][1][1][1][1][1][1][1][1][1][1][2] [1][1][1][1][1][1][1][1][1][1][1][1][1][3] [1][1][1][1][1][1][1][1][1][1][1][1][1][1][4] [1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][5] [1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][6] [1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][7] [1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][8] [0][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1] [1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1] [2][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1] [3][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1] [4][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1] [5][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1] [6][1][1][1][1][1][1][1][1][1][1][1][1][1][1][1] [7][1][1][1][1][1][1][1][1][1][1][1][1][1][1] [8][1][1][1][1][1][1][1][1][1][1][1][1][1] [9][1][1][1][1][1][1][1][1][1][1][1][1] [0][1][1][1][1][1][1][1][1][1][1][1] [1][1][1][1][1][1][1][1][1][1] [2][1][1][1][1][1][1][1][1] [3][1][1][1][1][1][1][1] [4][1][1][1][1][1][1] [5][1][1][1][1][1] [6][1][1][1][1] [7][1][1][1] [8][1][1] [9]

[PT] arrays

O - Quadrantes

Escreva um programa que leia uma matriz quadrada de números inteiros e visualize a soma dos números de cada um dos seus quadrantes.
Se dividir-mos ao meio uma matriz na vertical e na horizontal obtemos 4 quadrantes (q1, q2, q3, q4) na seguinte ordem:

q2	q1
q3	q4

Cada quadrante possui elementos próprios, não partilhados com outros quadrantes.

O programa deve ler um valor N correspondendo à dimensão da matriz (1 < N <= 20). De seguida, deve ler N linhas contendo, cada uma, N números separados por espaços.

No final deve ser visualizado o valor dos 4 quadrantes da seguinte forma:

[q2][q1]
[q3][q4]

[EN] arrays

O - Quadrants

Write a program that reads a square matrix of integers and displays the sum of the numbers in each of its quadrants.
If we divide a matrix in half vertically and horizontally, we obtain 4 quadrants (q1, q2, q3, q4) in the following order:

q2	q1
q3	q4

Each quadrant has its own elements, not shared with other quadrants.

The program must read a value N corresponding to the dimension of the array (1 < N <= 20). Then it should read N lines, each containing N numbers separated by spaces.

At the end, the value of the 4 quadrants should be visualized as follows:

[q2][q1]
[q3][q4]

Example1:

Input	Output
3 10 20 30 50 60 70 90 100 110	[10][30] [90][110]

Example2:

Input	Output
4 10 20 30 40 50 60 70 10 90 100 110 1 1 2 3 4	[140][150] [193][118]

Example3:

Input	Output
2 10 20 30 40	[10][20] [30][40]

Example4:

Input	Output
20 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 19 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20	[550][550] [1550][1550]

Example5:

Input	Output
19 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 9 9 9 9 9 9 9 9 2 9 9 9 9 9 9 9 9 9 1 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 13 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14	[398][397] [1198][1197]

15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 19 19 19 19 19 19 19 2 19 19 19 19 19 19 19 19 19 19 1	

[PT] arrays

P - Fatorial dos algarismos

Escreva um programa que leia um número inteiro N (0 <= N <= 10000000) e um inteiro X (0 <= X <= 10000000) e determine quantos dos números positivos de zero até N (inclusive) possuem a soma dos fatoriais dos seus algarismos superior a X.

Implemente uma solução eficiente.

[EN] arrays

P - Factorial of digits

Write a program that reads an integer N (0 <= N <= 10000000) and an integer X (0 <= X <= 10000000) and determines how many of the positive numbers from zero to N (inclusive) have the sum of the factorials of the its digits greater than X.

Implement an efficient solution.

example: for N=5 and X=10
0 : 0! =1
1 : 1! =1
2 : 2! =2
3 : 3! =6
4 : 4! =24
5 : 5! =120
output: 2

Example1:

Input	Output
5 10	2

Example2:

Input	Output
100000000 55555	67047520

Example3:

Input	Output
100000000 100000000	0

Example4:

Input	Output
1000 10000	510

[PT] arrays

Q - Condomínio (v2)

Considere um prédio de apartamentos com E entradas e F pisos, completamente ocupados.
Faça um programa modular que:
-Primeiro, leia a quantidade de entradas (E) e de pisos (F) existentes (dois inteiros na mesma linha, separados por um espaço).
-Segundo, leia o nome do proprietário de cada um dos apartamentos. A leitura deve ser feita por ordem cresente

de entrada e piso, isto é, para cada entrada devem ser lidos os proprietáios dos respetivos pisos.
-De seguida, o programa deve ler um dado nome inserido pelo utilizador, procurar e mostrar para esse nome qual é a respetiva entrada e piso, no seguinte formato:

"name=<nome>"
"entrance=<nº da entrada>"
"floor=<nº do piso>"

Se existirem múltiplas soluções (vários proprietários com o mesmo nome), deve indicar todas as possibilidades, por ordem ascendente de entrada e piso.
Caso não exista um proprietário com esse nome no prédio, deve imprimir a mensagem **"Do not live in the building"**.

[EN] arrays

Q - Condominium (v2)

Consider a completely occupied apartment building with E entrances and F floors.
Write a modular program to:
-First, read the number of entrances (E) and floors (F) that exist (two integers on the same line, separated by a space).
-Second, read the name of the owner of each apartment. The reading should be done in order of entrance and floor, ie, for each entrance must be read the owners of each of its floors;
-Next, read a particular name and display which entrance and floor it owns, in the following format:

"name=<name>"
"entrance=<entrance number>"
"floor=<floor number> "

If there are multiple solutions (multiple owners with the same name), all possibilities should be displayed, in ascending order of entrance and floor.
If there is no owner with that name in the building, the message **"Do not live in the building"** should be displayed.

Example:

floor

3	Luisa Lima	Hercilia Hora	Maria Mota
2	Carla Costa	Gloria Gomes	Luisa Lima
1	Berta Barata	Francisca Fe	Joana Jales
0	Ana Anacleta	Elsa Eira	Ines Iris
	0	1	2

entrance

Example1: (search for Luisa Lima)

Input	Output
3 4 Ana Anacleta Berta Barata Carla Costa Luisa Lima Elsa Eira Francisca Fé Gloria Gomes Hercilia Hora Ines Iris Joana Jales Luisa Lima Maria Mota **Luisa Lima**	name=Luisa Lima entrance=0 floor=3 name=Luisa Lima entrance=2 floor=2

Example 2: (search for Zita Zulmira)

Input	Output
3 4 Ana Anacleta Berta Barata Carla Costa Luisa Lima Elsa Eira Francisca Fe Gloria Gomes Hercilia Hora Ines Iris	Do not live in the building

Joana Jales Luisa Lima Maria Mota **Zita Zulmira**	

Example 3: (search for Luisa)

Input	Output
3 4 Ana Berta Barata Carla Luisa Lima Elsa Francisca Fé Gloria Hercilia Hora Ines Joana Jales Luisa Maria Mota **Luisa**	name=Luisa entrance=2 floor=2

Example 4: (search for Mario M Mota)

Input	Output
3 1 Mario M Mota Mario M Mota Mario M Mota **Mario M Mota**	name=Mario M Mota entrance=0 floor=0 name=Mario M Mota entrance=1 floor=0 name=Mario M Mota entrance=2 floor=0

Example 5: (search for Mario M Mota)

Input	Output
1 3 Mario M Mota Maria M Mata Mario M Mota **Mario M Mota**	name=Mario M Mota entrance=0 floor=0 name=Mario M Mota entrance=0 floor=2

Example 6: (search for Mario M Mota)

Input	Output
1 1 Mario M Mota **Mario M Mota**	name=Mario M Mota entrance=0 floor=0

Example 7: (search for Mario S Mota)

Input	Output
1 1 Mario M Mota **Mario S Mota**	Do not live in the building