

**FORM KESEPAKATAN SOAL UJIAN ~~D3~~ / ~~Eks. D3~~ → S1 / S1 / S2 (\*)**  
**Semester Genap / Ganjil (\*) 2024/2025**

**Kode – Nama Dosen** : D5337 — Yudy Purnama S.Kom., M.T.I.  
D5387 — Harvianto, S.Kom., M.T.I.  
D6957 — Douglas Rakasiwi Nugroho, S.Kom.,  
M.Kom.

**Kode – Nama Mata kuliah** : COMP6047001— Algorithm and Programming

**Durasi Ujian** : 200 menit

**Sifat Ujian** : ~~Buka Buku~~ / Tutup Buku

**Buku Ujian** : .....

**Supporting Tools** : .....

**Penggandaan Supporting File** : .....

**Learning Outcomes :**

LO1: (C2) Comprehension: explain kind of algorithms in problem solving  
LO2: (C3) Application: apply syntax and functions in C language in problem solving  
LO3: (C3) Application: construct a program using C language in problem solving  
LO4: (C3) Application: design a program with file processing using C language in problem solving  
LO5: (C6) Evaluation: choose the best sorting and searching algorithm in problem solving

No	Tipe Soal (Essay/Kasus)	Bobot (%)	LO Terkait	Topik/Materi yang Diujikan
1	Kasus	20%	2, 3	Recursive
2	Kasus	20%	2, 3, 5	Sorting
3	Kasus	20%	2, 3, 5	Sorting
4	Kasus	20%	2, 3, 5	Sorting
5	Kasus	20%	2, 3, 4	File processing

Jakarta, 4 September 2024

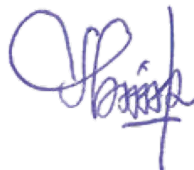
Dibuat oleh,



(D5337 — Yudy Purnama  
S.Kom., M.T.I.)

**Dosen Pembuat Soal**

Dicek oleh,



(D7018 - Shania Priccilia,  
S.Kom., M.Kom.)  
**Dosen SCC**

Disetujui oleh,

(D5874 – Irene Anindaputri  
Iswanto, S.Kom., M.Sc.Eng.)  
**Head of Computer Science  
Program**

## BINUS University

<b>Academic Career:</b> <i>Undergraduate / <del>Master</del> / <del>Doctoral</del> *)</i>		<b>Class Program:</b> <i><del>International</del> / <del>Regular</del> / <del>Smart Program</del> / <del>Global Class</del> / <del>BINUS Online Learning</del> *)</i>	
<input type="checkbox"/> Mid Exam <input type="checkbox"/> Compact Term Exam <input checked="" type="checkbox"/> Final Exam <input type="checkbox"/> Others Exam : _____		<b>Term : <del>Odd</del> / Even / <del>Compact</del> *)</b> <b>Period (Only for BOL) : 1 / 2 *)</b>	
<input checked="" type="checkbox"/> Kemanggisian <input type="checkbox"/> Senayan <input type="checkbox"/> Semarang <input type="checkbox"/> Alam Sutera <input type="checkbox"/> Bandung <input type="checkbox"/> Bekasi <input type="checkbox"/> Malang		<b>Academic Year :</b> <b>2024 / 2025</b>	
Exam Type* : Onsite / <del>Online</del>		Faculty / Dept. : School of Computer Science	
Day / Date** :		Code - Course : COMP6047001— Algorithm and Programming	
Time** :		Code - Lecturer : D5337 — Yudy Purnama S.Kom., M.T.I. D5387 — Harvianto, S.Kom., M.T.I. D6957 — Douglas Rakasiwi Nugroho, S.Kom., M.Kom.	
Exam Specification*** : <input type="checkbox"/> Open Book <input type="checkbox"/> Open Notes <input checked="" type="checkbox"/> Close Book <input type="checkbox"/> Submit Project <input type="checkbox"/> Open E-Book <input type="checkbox"/> Oral Test		BULC (Only for BOL) : -	
Equipment*** : <input type="checkbox"/> Exam Booklet <input type="checkbox"/> Laptop <input type="checkbox"/> Drawing Paper – A3 <input type="checkbox"/> Calculator <input type="checkbox"/> Tablet <input type="checkbox"/> Drawing Paper – A2 <input type="checkbox"/> Dictionary <input type="checkbox"/> Smartphone <input type="checkbox"/> Notes		Student ID *** :	
		Name *** :	
		Signature *** :	
□) <i>Strikethrough the unnecessary items</i> **) <i>For Online Exam, this is the due date</i> ***) <i>Only for Onsite Exam</i>			
<b><i>Please insert the test paper into the exam booklet and submit both papers after the test.</i></b>  <b><i>The penalty for CHEATING is DROP OUT!</i></b>			

### Learning Outcomes:

- LO1: (C2) Comprehension: explain kind of algorithms in problem solving
- LO2: (C3) Application: apply syntax and functions in C language in problem solving
- LO3: (C3) Application: construct a program using C language in problem solving
- LO4: (C3) Application: design a program with file processing using C language in problem solving
- LO5: (C6) Evaluation: choose the best sorting and searching algorithm in problem solving

## 1. (20 points) [English] As the Temperature Dictates

In a bustling robotic utopia, countless robots operated tirelessly, each designed to function optimally within specific temperature ranges. The city's central AI, Hov, faced a crucial challenge: efficiently assigning robots to rooms.

Hov decided to organize the office into several rooms, and for each room, the temperature would be set to a single value, chosen based on the robots assigned to that room. The temperature of each room can be set to any value, but once set, it remains fixed for the robots in that room. Each room can hold any number of robots, as long as the room's temperature falls within the preferred temperature range for each robot assigned to it.

Hov tasked its programmers with developing a solution: a program that determines the minimum number of rooms needed to accommodate all robots, while respecting their temperature preferences. The fate of the city hinges on this task—will you rise to the challenge and find each robot its perfect room?

### Input Format

The first line contains an integer  $N$ , indicating the number of robots needs to be allocated. Followings are  $N$  lines of each a pair of integers ( $T1$  and  $T2$ ) indicating each robot temperature preference range.

### Output Format

The output will an integer  $R$ , which is the minimum number of rooms required to accommodate all the robot's temperature preferences.

### Constraints

- $1 \leq N \leq 10^4$
- $0 \leq T1, T2 \leq 10^5$
- $T1 \leq T2$

### Sample Input 1

```
5
1 3
1 2
2 3
1 1
4 5
```

### Sample Output 1

```
3
```

### Explanation of Test Case 1

Three rooms will be sufficient. The three robots with preferences (1–1), (1–2), and (1–3) can share one room, as their temperature ranges overlap. The other two robots, with preferences (2–3) and (4–5), will each need their own separate rooms.

### Sample Input 2

```
5
1 3
2 4
5 7
8 10
11 12
```

### Sample Output 2

```
4
```

### Explanation of Test Case 2

There should be at least four different rooms. The robots with temperature range preferences (1–3) and (2–4) can share one room, while the other robots with preferences (5–7), (8–10), and (11–12) must each have their own separate room.

## **(20 points) [Indonesian] As the Temperature Dictates**

Di sebuah utopia robot yang sibuk, tak terhitung jumlahnya robot yang bekerja tanpa lelah, masing-masing dirancang untuk berfungsi secara optimal dalam rentang suhu tertentu. AI pusat kota, Hov, menghadapi tantangan besar: menetapkan penugasan robot ke kamar dengan efisien. Hov memutuskan untuk membagi kantor menjadi beberapa ruangan, dan untuk setiap ruangan, suhu akan diatur ke sebuah nilai yang dipilih. Suhu setiap ruangan dapat diatur ke berapapun nilai yang diinginkan. Setiap kamar dapat menampung sebanyak mungkin robot, selama suhu ruangan tersebut berada dalam rentang suhu yang disukai oleh masing-masing robot yang ditugaskan ke dalamnya.

Hov menugaskan para pemrogramnya untuk mengembangkan solusi, berupa sebuah program yang menentukan jumlah ruangan minimum yang dibutuhkan untuk menampung semua robot, sambil tetap memenuhi preferensi suhu mereka. Nasib kota bergantung pada tugas ini—mampukah Anda menghadapi tantangan ini dan menemukan ruangan yang sempurna untuk setiap robot?

### **Input Format**

Baris pertama berisi sebuah bilangan bulat  $N$  yang menunjukkan jumlah robot yang perlu ditempatkan. Baris-baris berikutnya sebanyak  $N$  masing-masing berisi sepasang bilangan bulat ( $T1$  dan  $T2$ ) yang menunjukkan rentang suhu preferensi setiap robot.

### **Output Format**

Output akan berupa sebuah bilangan bulat  $R$ , yang merupakan jumlah minimum ruangan yang diperlukan untuk menampung semua preferensi suhu robot.

### **Constraints**

- $1 \leq N \leq 10^3$
- $0 \leq T1, T2 \leq 10^5$
- $T1 \leq T2$

### Sample Input 1

```
5
1 3
1 2
2 3
1 1
4 5
```

### Sample Output 1

```
3
```

### Explanation of Test Case 1:

Tiga kamar sudah cukup. Tiga robot dengan preferensi rentang suhu (1–1), (1–2), dan (1–3) dapat berbagi satu kamar, karena rentang suhu mereka saling beririsan. Dua robot lainnya, dengan preferensi (2–3) dan (4–5), masing-masing membutuhkan kamar terpisah.

### Sample Input 2

```
5
1 3
2 4
5 7
8 10
11 12
```

### Sample Output 2

```
4
```

### Explanation of Test Case 2

Harus ada setidaknya empat kamar yang berbeda. Dua robot dengan preferensi rentang suhu (1–3) dan (2–4) dapat berbagi satu kamar, sementara robot-robot lainnya dengan preferensi (5–7), (8–10), dan (11–12) masing-masing harus memiliki kamar terpisah.

## 2. (20 points) [English] Birthdate

You are given a list of people, where each person is represented by their name and date of birth. Your task is to sort this list in ascending order, first by date of birth and then by name if two or more people have the same date of birth. The date of birth is in the format "YYYY-MM-DD", where "YYYY" is the year, "MM" is the month, and "DD" is the day. The list of people is indexed starting from 1. You are also given a number  $Q$ , which is an integer between 1 and 20000. For each query, you are given an integer  $X$ , which is between one and  $N$ , where  $N$  is the number of people. You must print the name of the person at index  $X$  in the sorted list.

You will get a maximum of 16 points if you use slower sorting algorithms such as bubble sort, selection sort, or insertion sort. These algorithms have higher time complexity and are less efficient for large datasets. To achieve full points, you should use more efficient sorting algorithms like quick sort or merge sort.

### Input Format

The first line of input contains an integer  $N$ , which is the number of people. The next  $N$  lines contain the names and dates of birth of the people. Each line contains a date  $D$  and a string  $S$ , separated by a space.  $S$  is the name of the person. The next line contains an integer  $Q$ , which is the number of queries. The next  $Q$  lines contain an integer  $X$ .

### Output Format

For each query, print a line containing the date of birth and the name of the person at index  $X$  in the sorted list.

### Constraints

- $1 \leq N \leq 20000$
- $1 \leq \text{length of } S \leq 20$
- $S$  contains only uppercase letters
- $1 \leq Q \leq 20000$
- $1 \leq X \leq N$

```
10
2004-09-18 NATHANIEL
2004-05-22 TANWIPUTRA
2003-09-11 ANGER
2005-02-25 VINCENT
2004-07-29 ADRIEL
2004-03-29 RIO
2003-03-04 OLIVIA
2003-09-16 THEODORUS
2003-09-01 NYOMAN
2005-04-05 CLARISSA
3
1
2
3
```

### Sample Output 1

```
2003-03-04 OLIVIA
2003-09-01 NYOMAN
2003-09-11 ANGER
```

### Explanation of Test Case 1

The list of people sorted by their dates of birth is as follows:

1. 2003-03-04 OLIVIA
2. 2003-09-01 NYOMAN
3. 2003-09-11 ANGER
4. 2003-09-16 THEODORUS
5. 2004-03-29 RIO
6. 2004-05-22 TANWIPUTRA
7. 2004-07-29 ADRIEL
8. 2004-09-18 NATHANIEL
9. 2005-02-25 VINCENT
10. 2005-04-05 CLARISSA



## Sample Input 2

```
10
2005-03-29 FANCE
2005-11-30 JIOVANNI
2003-04-13 CORNELLIUS
2004-08-22 TIMOTIUS
2005-03-29 PRATAMA
2005-02-08 GABRIEL
2005-09-25 SIDNEY
2004-09-08 SAMUDRA
2003-10-18 AGUNG
2005-10-07 KENT
5
2
4
6
8
10
```

## Sample Output 2

```
2003-10-18 AGUNG
2004-09-08 SAMUDRA
2005-03-29 FANCE
2005-09-25 SIDNEY
2005-11-30 JIOVANNI
```

## Explanation of Test Case 2

The list of people sorted by their dates of birth is as follows:

1. 2003-04-13 CORNELLIUS
2. 2003-10-18 AGUNG
3. 2004-08-22 TIMOTIUS
4. 2004-09-08 SAMUDRA
5. 2005-02-08 GABRIEL
6. 2005-03-29 FANCE
7. 2005-03-29 PRATAMA
8. 2005-09-25 SIDNEY
9. 2005-10-07 KENT
10. 2005-11-30 JIOVANNI

## (20 points) [Indonesian] Birthdate

Anda diberikan sebuah daftar orang, setiap orang direpresentasikan dengan nama dan tanggal lahir mereka. Tugas Anda adalah mengurutkan daftar ini dalam urutan *ascending*, pertama berdasarkan tanggal lahir, dan jika ada dua atau lebih orang dengan tanggal lahir yang sama, urutkan berdasarkan nama mereka. Tanggal lahir diberikan dalam format "YYYY-MM-DD", di mana "YYYY" adalah tahun, "MM" adalah bulan, dan "DD" adalah hari. Daftar orang diberi indeks mulai dari one. Anda juga diberikan sebuah bilangan  $Q$ , yaitu sebuah bilangan bulat antara 1 dan 20000. Untuk setiap query, Anda diberikan sebuah bilangan bulat  $X$ , yang berada di antara 1 dan  $N$ , di mana  $N$  adalah jumlah orang dalam daftar. Anda harus mencetak nama orang pada indeks  $X$  dalam daftar yang telah diurutkan.

Anda akan mendapatkan maksimum 16 poin jika menggunakan algoritma *sorting* yang lambat seperti *bubble*, *selection*, atau *insertion* sort. Algoritma tersebut memiliki kompleksitas waktu yang lebih tinggi dan kurang efisien untuk dataset besar. Untuk mendapatkan poin penuh, Anda harus menggunakan algoritma *sorting* yang lebih efisien seperti *quick* atau *merge* sort.

### Input Format

Baris pertama input adalah sebuah bilangan bulat  $N$ , yang menunjukkan jumlah orang. Berikutnya  $N$  baris berisi nama dan tanggal lahir orang-orang tersebut. Setiap baris berisi sebuah tanggal  $D$  dan sebuah string  $S$ , dipisahkan sebuah spasi.  $S$  adalah nama orang tersebut. Baris berikutnya berisi sebuah bilangan bulat  $Q$ , yang menunjukkan jumlah *query*. Berikutnya adalah  $Q$  baris berisi sebuah bilangan bulat  $X$ .

### Output Format

Untuk setiap *query*, tampilkan sebuah baris berisi tanggal dan nama orang yang berada di posisi  $X$  setelah diurutkan.

### Constraints

- $1 \leq N \leq 20000$
- $1 \leq \text{panjang dari } S \leq 20$
- $S$  hanya berisi huruf *uppercase*
- $1 \leq Q \leq 20000$
- $1 \leq X \leq N$

### Sample Input 1

```
10
2004-09-18 NATHANIEL
2004-05-22 TANWIPUTRA
2003-09-11 ANGER
2005-02-25 VINCENT
2004-07-29 ADRIEL
2004-03-29 RIO
2003-03-04 OLIVIA
2003-09-16 THEODORUS
2003-09-01 NYOMAN
2005-04-05 CLARISSA
3
1
2
3
```

### Sample Output 1

```
2003-03-04 OLIVIA
2003-09-01 NYOMAN
2003-09-11 ANGER
```

### Explanation of Test Case 1

Daftar orang setelah diurutkan berdasarkan tanggal lahir adalah sebagai berikut:

1. 2003-03-04 OLIVIA
2. 2003-09-01 NYOMAN
3. 2003-09-11 ANGER
4. 2003-09-16 THEODORUS
5. 2004-03-29 RIO
6. 2004-05-22 TANWIPUTRA
7. 2004-07-29 ADRIEL
8. 2004-09-18 NATHANIEL
9. 2005-02-25 VINCENT
10. 2005-04-05 CLARISSA

## Sample Input 2

```
10
2005-03-29 FANCE
2005-11-30 JIOVANNI
2003-04-13 CORNELLIUS
2004-08-22 TIMOTIUS
2005-03-29 PRATAMA
2005-02-08 GABRIEL
2005-09-25 SIDNEY
2004-09-08 SAMUDRA
2003-10-18 AGUNG
2005-10-07 KENT
5
2
4
6
8
10
```

## Sample Output 2

```
2003-10-18 AGUNG
2004-09-08 SAMUDRA
2005-03-29 FANCE
2005-09-25 SIDNEY
2005-11-30 JIOVANNI
```

## Explanation of Test Case 2

Daftar data orang setelah diurutkan berdasarkan tanggal lahir adalah sebagai berikut:

1. 2003-04-13 CORNELLIUS
2. 2003-10-18 AGUNG
3. 2004-08-22 TIMOTIUS
4. 2004-09-08 SAMUDRA
5. 2005-02-08 GABRIEL
6. 2005-03-29 FANCE
7. 2005-03-29 PRATAMA
8. 2005-09-25 SIDNEY
9. 2005-10-07 KENT
10. 2005-11-30 JIOVANNI

### 3. (20 points) [English] Decrypted Listmates

There is currently a trend among Rino's friends to create a sorted list of close friends in a file named *"testdata.in"*. One day, Rino wants to check if a name is included in his friends' lists. Unfortunately, each name on the list is encrypted. The encryption method involves shifting the ASCII values of certain characters. Please figure it out yourself based on the given example. Help Rino determine whether his name is on the list or not.

#### Input Format

Read from file *"testdata.in"*. The first line consists of one integer  $N$ , which is the number of the owner's best friends. The next  $N$  lines consist of encrypted name  $S$ . The last line is a string  $A$ , which is a name to be searched.

#### Output Format

The output is either "Yes" or "No". "Yes" if the name is in the list, and "No" if not.

#### Constraints

- $1 \leq N \leq 10$
- $0 \leq |S|, |A| \leq 99$

#### Sample Input 1

```
5
Jfnoif
Jjspo
Ljsb
Rjnp
Rpsf
Rino
```

#### Sample Output 1

```
Yes
```

### Explanation of Test Case 1

The result of the decryption is:

```
Jennie  
Jisoo  
Lisa  
Rino  
Rose
```

Therefore, the name Rino is present and the program outputs Yes.

### Sample Input 2

```
9  
Ciafypuog  
Dbhzuo  
Jfoogzepn  
Jjhzo  
Mjnb  
Mpmp  
Nbyfoo  
Sbnb  
T{uzu  
Karina
```

### Sample Output 2

```
No
```

### Explanation of Test Case 2

The result of the decryption is:

```
Chaeyoung  
Dahyun  
Jeongyeon  
Jihyo  
Mina  
Momo  
Nayeon  
Sana  
Tzuyu
```

Therefore, the name Karina is not present and the program outputs No.

## (20 points) [Indonesian] Decrypted Listmates

Saat ini ada tren di kalangan teman-teman Rino untuk membuat daftar terurut nama teman dekat dalam sebuah file bernama *"testdata.in"*. Suatu hari, Rino ingin memeriksa apakah suatu nama ada dalam daftar nama teman-temannya. Sayangnya, setiap nama dalam daftar terenkripsi. Metode enkripsi melibatkan pergeseran nilai ASCII dari karakter-karakter tertentu. Silakan temukan sendiri berdasarkan contoh yang diberikan. Bantu Rino untuk menentukan apakah nama tersebut ada dalam daftar tersebut atau tidak.

### Input Format

Baca input dari file *"testdata.in"*. Baris pertama terdiri dari satu bilangan bulat  $N$ , yang merupakan jumlah teman terbaik pemiliknya.  $N$  baris berikutnya berisi  $S$ , yaitu nama yang terenkripsi. Baris terakhir merupakan  $A$ , yaitu nama yang ingin dicari.

### Output Format

Outputnya adalah "Yes" atau "No". "Yes" jika nama Rino ada dalam daftar, dan "No" jika nama Rino tidak ada dalam daftar.

### Constraints

- $1 \leq N \leq 10$
- $0 \leq |S|, |A| \leq 99$

### Sample Input 1

```
5
Jfnoif
Jjspo
Ljsb
Rjnp
Rpsf
Rino
```

### Sample Output 1

```
Yes
```

### Explanation of Test Case 1

Hasil dari proses dekripsi adalah:

```
Jennie  
Jisoo  
Lisa  
Rino  
Rose
```

Jadi, nama Rino ada di dalam list dan program menampilkan Yes.

### Sample Input 2

```
9  
Ciafypuog  
Dbhzuo  
Jfoogzepn  
Jjhzo  
Mjnb  
Mpmp  
Nbyfoo  
Sbnb  
T{uzu  
Karina
```

### Sample Output 2

```
No
```

### Explanation of Test Case 2

Hasil dari proses dekripsi adalah:

```
Chaeyoung  
Dahyun  
Jeongyeon  
Jihyo  
Mina  
Momo  
Nayeon  
Sana  
Tzuyu
```

Jadi, nama Karina tidak ada di dalam list dan program menampilkan No.



#### 4. (20 points) [English] Double Ladder

In a small village, a craftsman is building a unique structure that requires two ladders. The lengths of these ladders must add up to exactly the target length specified for the project. The craftsman has a list of available ladder lengths, and your task is to help him find the two lengths that meet this requirement.

##### Input Format

The first line consists of an integer  $N$ , which indicates the number of available ladders. The next line consists of  $N$  integers, representing the length of each ladder. The last line consists of an integer  $X$ , which is the target length.

##### Output Format

The output is "Yes" or "No". "Yes" if there are two ladder lengths whose sum equals the target. If not, display "No".

##### Constraints

- $2 \leq N \leq 10^5$
- $0 \leq A, X \leq 10^6$

##### Sample Input 1

```
5
4 7 2 8 6
10
```

##### Sample Output 1

```
Yes
```

##### Explanation of Test Case 1

The possible combination of pairing would be (4, 6) and (2, 8).

### Sample Input 2

```
6
10 5 14 19 7 2
20
```

### Sample Output 2

```
No
```

### Explanation of Test Case 2

There is no possible ladder pairing available.

### (20 points) [Indonesian] Double Ladder

Di sebuah desa kecil, seorang tukang sedang membangun struktur unik yang memerlukan dua buah tangga. Gabungan panjang kedua tangga tersebut harus tepat sama dengan panjang target yang ditentukan untuk proyek tersebut. Tukang tersebut memiliki daftar panjang tangga yang tersedia, dan tugas Anda adalah membantunya menemukan dua tangga yang memenuhi syarat ini.

#### Input Format

Baris pertama terdiri dari sebuah bilangan bulat  $N$  yang menunjukkan jumlah tangga yang tersedia. Baris berikutnya terdiri dari  $N$  bilangan bulat yang menunjukkan panjang masing-masing tangga. Dan baris terakhir terdiri dari sebuah bilangan bulat  $X$  yang merupakan panjang target.

#### Output Format

Outputnya adalah "Yes" atau "No". "Yes" jika terdapat dua panjang tangga yang jumlahnya sama dengan target. Jika tidak, tampilkan "No".

#### Constraints

- $2 \leq N \leq 10^5$
- $0 \leq A, X \leq 10^6$

#### Sample Input 1

```
5
4 7 2 8 6
10
```

#### Sample Output 1

```
Yes
```

#### Explanation of Test Case 1

Beberapa kombinasi pasangan yang memenuhi adalah (4, 6) dan (2, 8).

### Sample Input 2

```
6
10 5 14 19 7 2
20
```

### Sample Output 2

```
No
```

### Explanation of Test Case 2

Tidak ada pasangan yang tersedia.

## 5. (20 points) [English] Harvester

A harvester is a machine that collects crops. In this problem, you are given a map of a field with crops and harvesters. There are  $M$  harvesters, and each harvester is represented by a unique letter from 'a' to 'z'. The map is a square grid of size  $N \times N$ , where  $N$  is an integer between 1 and 100. Each cell in the grid contains either:

- A digit between 0 and 9, representing the number of crops in that cell.
- A character between 'a' and 'z', representing a harvester.
- A symbol '#', representing an obstacle that the harvester cannot move through.

Each harvester can move up, down, left, or right, but not diagonally. When a harvester moves to a cell, it collects the crops in that cell and all the cells that are reachable from it. If a crop is reachable, it is guaranteed to be reachable by only one harvester. The harvester cannot move through obstacles or revisit a cell. Your task is to calculate the total amount of crops collected by each harvester.

### Input Format

The first line of input contains two integers  $N$  and  $M$ , separated by a space.  $N$  is the size of the field and  $M$  is the number of harvesters. The next  $N$  lines contain the map of the field. Each line contains  $N$  characters.

### Output Format

For each harvester, in ascending order by name, print a line containing the character representing the harvester followed by the total amount of crops collected by that harvester, separated by a space.

### Constraints

- $1 \leq N \leq 100$
- $1 \leq M \leq 26$

5 4  
69767  
7711#  
7#b#2  
#c#0d  
a#665

### Sample Output 1

a 0  
b 58  
c 0  
d 19

### Explanation of Test Case 1

6	9	7	6	7
7	7	1	1	#
7	#	<b>b</b>	#	2
#	<b>c</b>	#	0	<b>d</b>
<b>a</b>	#	6	6	5

- The harvester 'a' cannot move anywhere.
- The harvester 'b' can collect  $6 + 9 + 7 + 6 + 7 + 7 + 7 + 1 + 1 + 7 = 58$ .
- The harvester 'c' cannot move anywhere.
- The harvester 'd' can collect  $2 + 0 + 6 + 6 + 5 = 19$ .

## Sample Input 2

```
10 6
299#435#86
4#2#7##050
5#e#6#9679
#b##8#1553
1#714#775#
c9##6#02#4
4#49f#0#71
3#4##4d#a5
7#179##771
79###01148
```

## Sample Output 2

```
a 46
b 0
c 40
d 89
e 31
f 85
```

## Explanation of Test Case 2

2	9	9	#	4	3	5	#	8	6
4	#	2	#	7	#	#	0	5	0
5	#	e	#	6	#	9	6	7	9
#	b	#	#	8	#	1	5	5	3
1	#	7	1	4	#	7	7	5	#
c	9	#	#	6	#	0	2	#	4
4	#	4	9	f	#	0	#	7	1
3	#	4	#	#	4	d	#	a	5
7	#	1	7	9	#	#	7	7	1
7	9	#	#	#	0	1	1	4	8

- Harvester 'a' can collect  $4 + 7 + 1 + 5 + 7 + 7 + 1 + 0 + 1 + 1 + 4 + 8 = 46$ .
- Harvester 'b' can not move anywhere.
- Harvester 'c' can collect  $1 + 9 + 4 + 3 + 7 + 7 + 9 = 40$ .
- Harvester 'd' can collect  $8 + 6 + 0 + 5 + 0 + 9 + 6 + 7 + 9 + 1 + 5 + 5 + 3 + 7 + 7 + 5 + 0 + 2 + 0 + 4 = 89$ .
- Harvester 'e' can collect  $2 + 9 + 9 + 4 + 2 + 5 = 31$ .
- Harvester 'f' can collect  $4 + 3 + 5 + 7 + 6 + 8 + 7 + 1 + 4 + 6 + 4 + 9 + 4 + 1 + 7 + 9 = 85$ .

## (20 points) [Indonesian] Harvester

*Harvester* adalah mesin yang digunakan untuk mengumpulkan hasil panen. Dalam masalah ini, Anda diberikan peta sebuah ladang dengan beberapa hasil panen dan pemanen. Terdapat  $M$  *harvester*, dan setiap *harvester* direpresentasikan oleh sebuah huruf unik dari 'a' hingga 'z'. Peta tersebut berupa persegi berukuran  $N \times N$ , di mana  $N$  adalah bilangan bulat antara 1 dan 100. Setiap sel dalam grid berisi salah satu dari berikut:

- Sebuah digit antara 0 dan 9, yang merepresentasikan jumlah hasil panen di sel tersebut.
- Sebuah karakter antara 'a' hingga 'z', yang merepresentasikan *harvester*.
- Sebuah simbol '#', yang merepresentasikan halangan yang tidak dapat dilalui oleh *harvester*.

Setiap *harvester* dapat bergerak ke atas, bawah, kiri, atau kanan, tetapi tidak secara diagonal. Ketika *harvester* bergerak ke sebuah sel, *harvester* tersebut akan mengumpulkan hasil panen di sel tersebut dan semua sel lain yang dapat dicapai dari sel itu. Jika suatu hasil panen dapat dicapai, dijamin hanya akan dicapai oleh satu *harvester* saja. *Harvester* tidak dapat melewati halangan atau mengunjungi kembali sel yang sama. Tugas Anda adalah menghitung total hasil panen yang dikumpulkan oleh masing-masing *harvester*.

### Input Format

Baris pertama input berisi dua bilangan bulat  $N$  dan  $M$  yang dipisahkan oleh spasi.  $N$  adalah ukuran ladang, dan  $M$  adalah jumlah *harvester*. Baris berikutnya sebanyak  $N$  berisi peta ladang. Setiap baris berisi  $N$  karakter.

### Output Format

Untuk setiap pemanen, dalam urutan *ascending* berdasarkan nama, cetak satu baris yang berisi karakter yang merepresentasikan *harvester*, diikuti oleh total hasil panen yang dikumpulkan oleh *harvester* tersebut. Karakter dan total hasil panen dipisahkan oleh spasi.

### Constraints

- $1 \leq N \leq 100$
- $1 \leq M \leq 26$



5 4  
69767  
7711#  
7#b#2  
#c#0d  
a#665

### Sample Output 1

a 0  
b 58  
c 0  
d 19

### Explanation of Test Case 1

6	9	7	6	7
7	7	1	1	#
7	#	<b>b</b>	#	2
#	<b>c</b>	#	0	<b>d</b>
<b>a</b>	#	6	6	5

- Harvester 'a' tidak dapat bergerak kemana pun.
- Harvester 'b' dapat mengumpulkan  $6 + 9 + 7 + 6 + 7 + 7 + 7 + 1 + 1 + 7 = 58$ .
- Harvester 'c' tidak dapat bergerak kemana pun.
- Harvester 'd' dapat mengumpulkan  $2 + 0 + 6 + 6 + 5 = 19$ .

```

10 6
299#435#86
4#2#7##050
5#e#6#9679
#b##8#1553
1#714#775#
c9##6#02#4
4#49f#0#71
3#4##4d#a5
7#179##771
79###01148

```

## Sample Output 2

```

a 46
b 0
c 40
d 89
e 31
f 85

```

## Explanation of Test Case 2

2	9	9	#	4	3	5	#	8	6
4	#	2	#	7	#	#	0	5	0
5	#	e	#	6	#	9	6	7	9
#	b	#	#	8	#	1	5	5	3
1	#	7	1	4	#	7	7	5	#
c	9	#	#	6	#	0	2	#	4
4	#	4	9	f	#	0	#	7	1
3	#	4	#	#	4	d	#	a	5
7	#	1	7	9	#	#	7	7	1
7	9	#	#	#	0	1	1	4	8

- Harvester 'a' dapat mengumpulkan  $4 + 7 + 1 + 5 + 7 + 7 + 1 + 0 + 1 + 1 + 4 + 8 = 46$ .
- Harvester 'b' tidak dapat bergerak kemana pun.
- Harvester 'c' dapat mengumpulkan  $1 + 9 + 4 + 3 + 7 + 7 + 9 = 40$ .
- Harvester 'd' dapat mengumpulkan  $8 + 6 + 0 + 5 + 0 + 9 + 6 + 7 + 9 + 1 + 5 + 5 + 3 + 7 + 7 + 5 + 0 + 2 + 0 + 4 = 89$ .
- Harvester 'e' dapat mengumpulkan  $2 + 9 + 9 + 4 + 2 + 5 = 31$ .
- Harvester 'f' dapat mengumpulkan  $4 + 3 + 5 + 7 + 6 + 8 + 7 + 1 + 4 + 6 + 4 + 9 + 4 + 1 + 7 + 9 = 85$ .