



[DSA WEEKLY CONTEST T8 2025]. TEST 6. SINH, QUAY LUI

[Info](#)[Statistics](#)[Rankings](#)[Participation](#)[Submissions](#)[Virtual join](#)[DSA](#)

[Thuật Toán Sinh]. Bài 31. Tổng các tập con

[Submit](#)**Time limit:** 1.0 / **Memory limit:** 256M**Point:** 100

Cho mảng **A[]** gồm **N** phần tử, bạn hãy liệt kê tất cả các tổng khác nhau của các tập con khác rỗng của mảng **A[]**.

Ví dụ A[] = {1, 2, 3} có thể tạo thành các tổng 1, 2, 3, 4, 5, 6

Đầu vào

Dòng đầu tiên là số **N**

Dòng 2 gồm **N** số trong mảng **A[]**

Giới hạn

 $1 \leq N \leq 20$ $1 \leq A[i] \leq 10^9$

Đầu ra

In ra các tổng khác nhau theo thứ tự tăng dần

Ví dụ :



3

1 3 5

Copy

Output 01

1 3 4 5 6 8 9

Copy

[Thuật Toán Sinh]. Bài 32. Cụm từ bí mật passphrase

Submit

Time limit: 1.0 / **Memory limit:** 256M**Point:** 100

Passphrase là cụm từ bí mật được sử dụng trong các ví tiền điện tử hiện nay, biết được cụm từ bí mật của 1 ví tiền điện tử thì bạn có thể lấy tất cả tiền điện tử trong ví này. Hiện nay có các bạn trẻ Việt Nam bán tool dò **Passphrase**, họ sẽ sinh ra 12 từ ngẫu nhiên từ 2048 từ trong bộ từ **BIP39**.

Bây giờ **28tech** muốn bạn triển khai 1 tool dò **Passphrase**, bạn hãy sinh ra tất cả các bộ cụm từ gồm 6 từ ngẫu nhiên từ **N** từ cho trước. Bạn cần liệt kê các cụm **Passphrase** theo thứ tự từ điển tăng dần

Đầu vào

Dòng đầu tiên là số **N**

Dòng 2 gồm **N** từ trong bộ từ **BIP39**

Giới hạn

 $1 \leq N \leq 10$

Đầu ra

In ra các cụm **Passphrase** tạo được



Ví dụ :

Input 01

2

badge cart

Copy

Output 01

Copy



badge badge badge badge badge cart
badge badge badge badge cart badge
badge badge badge badge cart cart
badge badge badge cart badge badge
badge badge badge cart badge cart
badge badge badge cart cart badge
badge badge badge cart cart cart
badge badge cart badge badge badge
badge badge cart badge badge cart
badge badge cart badge cart badge
badge badge cart badge cart cart
badge badge cart cart badge badge
badge badge cart cart badge cart
badge badge cart cart cart badge
badge badge cart cart cart cart
badge cart badge badge badge badge
badge cart badge badge badge cart
badge cart badge badge cart badge
badge cart badge badge cart cart
badge cart badge cart badge badge
badge cart badge cart badge cart
badge cart badge cart cart badge
badge cart badge cart cart badge
badge cart badge cart cart cart
cart badge badge badge badge badge
cart badge badge badge badge cart
cart badge badge badge cart badge
cart badge badge badge cart cart
cart badge badge cart badge badge
cart badge badge cart badge cart
cart badge badge cart cart badge
cart badge badge cart cart cart
cart badge cart badge badge badge
cart badge cart badge badge cart



```

cart badge cart cart badge badge
cart badge cart cart badge cart
cart badge cart cart cart badge
cart badge cart cart cart cart
cart cart badge badge badge badge
cart cart badge badge badge cart
cart cart badge badge cart badge
cart cart badge badge cart cart
cart cart badge cart badge badge
cart cart badge cart badge cart
cart cart badge cart cart badge
cart cart badge cart cart cart
cart cart cart badge badge badge
cart cart cart badge badge cart
cart cart cart badge cart badge
cart cart cart badge cart cart
cart cart cart cart badge badge
cart cart cart cart badge cart
cart cart cart cart cart badge
cart cart cart cart cart cart

```

[Thuật Toán Sinh]. Bài 33. Địa chỉ ví điện tử

[Submit](#)

Time limit: 1.0 / **Memory limit:** 256M

Point: 100

28tech đang muốn phát triển đồng tiền điện tử là **28coin**, anh ta cần tạo ra các địa chỉ ví để gửi và nhận đồng tiền này. Mỗi địa chỉ ví là sự kết hợp của hoán vị các chữ cái từ 'a' tới **X** với **X** là chữ cái in thường cho trước với các tổ hợp chập **K** của **N** phần tử các số nguyên từ 1 tới **N**.

Nhiệm vụ của bạn là hãy liệt kê tất cả các địa chỉ có thể có.

Đầu vào

Dòng duy nhất chứa 2 số **N**, **K** và kí tự **X**



$1 \leq K \leq N \leq 15$

Đầu ra

In ra các địa chỉ ví

Ví dụ :

Input 01

5 3 c

Copy

Output 01

Copy



Hello, Nguyễn Năng Anh.

abc124
abc125
abc134
abc135
abc145
abc234
abc235
abc245
abc345
acb123
acb124
acb125
acb134
acb135
acb145
acb234
acb235
acb245
acb345
bac123
bac124
bac125
bac134
bac135
bac145
bac234
bac235
bac245
bac345
bca123
bca124
bca125
bca134
bca135
bca145
bca234
bca235
bca245
bca345
cab123
cab124



```
cab135
cab145
cab234
cab235
cab245
cab345
cba123
cba124
cba125
cba134
cba135
cba145
cba234
cba235
cba245
cba345
```

[Quay Lui - Nhánh Cận]. Bài 31. Đường đi ngắn nhất trên bảng số

[Submit](#)

Time limit: 1.0 / **Memory limit:** 256M

Point: 100

Cho một bảng số kích thước $N \times M$. Chi phí khi đi qua ô (i, j) bằng $A[i][j]$.

Nhiệm vụ của bạn là hãy tìm một đường đi từ ô $(1, 1)$ tới ô (N, M) sao cho chi phí là nhỏ nhất.

Tại mỗi ô, bạn được phép đi sang trái, sang phải, đi lên trên và xuống dưới.

Đầu vào

- Dòng 1 bắt đầu bởi hai số nguyên N và M .
- N dòng tiếp theo, mỗi dòng gồm M số nguyên $A[i][j]$

Giới hạn



- $0 \leq A[i][j] \leq 9$

Đầu ra

- In ra một số nguyên là chi phí nhỏ nhất cho đường đi tìm được.

Ví dụ :

Input 01

```
6 7
0 3 6 0 5 9 1
6 5 4 4 0 7 6
4 0 2 1 5 6 1
2 7 7 3 3 1 6
4 4 9 6 9 7 2
3 6 4 4 1 9 2
```

Copy

Output 01

```
28
```

Copy

[Quay Lui - Nhánh Cận]. Bài 32.
Đường đi ngắn nhất của quân
Mã

Submit

Time limit: 1.0 / **Memory limit:** 256M

Point: 100

Cho một bàn cờ vua kích thước $N \times N$. Chi phí khi đi qua ô (i,j) bằng $A[i][j]$.

Nhiệm vụ của bạn là hãy tìm một đường đi cho quân Mã từ ô (s, t) tới ô (u, v) sao cho chi phí là nhỏ nhất. Nếu quân Mã không thể đi tới ô đích thì hãy in ra -1.



- Dòng 1 bắt đầu bởi hai số nguyên **N** và **s, t, u, v**.
- **N** dòng tiếp theo, mỗi dòng gồm **N** số nguyên **A[i][j]**

Giới hạn

- $1 \leq N, M \leq 15$
- $0 \leq A[i][j] \leq 100$

Đầu ra

- In ra một số nguyên là chi phí nhỏ nhất cho đường đi tìm được.

Ví dụ :

Input 01

```
13
3 2 3 11
52 6 72 11 51 33 1 14 69 60 41 80 61
66 38 51 29 63 54 19 86 53 67 91 94 34
41 73 70 10 37 44 12 30 67 59 46 53 33
4 40 78 23 66 90 26 7 31 25 59 30 96
7 70 6 30 57 93 25 57 97 31 94 15 41
16 87 67 82 96 3 55 4 23 20 35 89 59
51 90 64 5 62 85 4 40 87 2 93 51 100
39 43 89 63 65 76 71 66 95 35 20 97 62
2 96 15 97 51 43 18 100 47 55 11 74 10
61 45 87 87 40 57 7 39 19 58 33 80 99
86 23 24 59 4 91 29 28 72 49 8 40 56
41 29 95 12 35 33 40 84 69 82 56 36 1
86 1 28 68 39 65 77 51 10 61 77 40 79
```

[Copy](#)

Output 01



[Quay Lui - Nhánh Cận]. Bài 33. Đường đi ngắn nhất của quân Mã 2

[Submit](#)

Time limit: 1.0 / **Memory limit:** 256M

Point: 100

Cho một bàn cờ vua kích thước $N \times N$. Chi phí khi đi qua ô (i, j) bằng $A[i][j]$.

Nhiệm vụ của bạn là hãy tìm một đường đi cho quân Mã từ ô (s, t) tới ô (u, v) sao cho chi phí là nhỏ nhất. Ngoài ra bạn cần liệt kê tất cả các đường đi có được chi phí nhỏ nhất đó.

Đầu vào

- Dòng 1 bắt đầu bởi hai số nguyên N và s, t, u, v .
- N dòng tiếp theo, mỗi dòng gồm N số nguyên $A[i][j]$

Giới hạn

- $1 \leq N, M \leq 15$
- $0 \leq A[i][j] \leq 10$

Đầu ra

- Dòng 1 in ra một số nguyên là chi phí nhỏ nhất cho đường đi tìm được.
- Các dòng tiếp theo mỗi dòng là được đi qua từng ô của con Mã, coi đường đi là một chuỗi ký tự, hãy in các đường đi theo thứ tự từ điển tăng dần.

Ví dụ :



14

Copy

13 8 4 3

5 4 8 9 9 5 3 8 1 9 3 2 6 7

3 3 8 6 2 1 10 1 10 6 7 9 7 4

6 6 7 5 1 8 4 5 6 4 4 8 10 4

4 7 2 1 6 7 8 10 6 6 8 7 6 3

8 10 5 2 5 2 7 8 1 2 7 9 4 4

4 4 1 10 5 6 10 4 3 4 4 10 7 1

2 7 8 10 3 5 3 8 2 7 1 9 6 2

7 8 4 9 1 4 1 10 8 8 5 2 1 8

2 8 1 5 8 1 4 6 7 9 7 5 4 10

3 7 3 8 5 7 5 1 5 1 3 2 5 8

7 3 6 2 8 2 3 2 3 6 4 10 2 2

3 3 1 5 5 5 6 8 3 2 3 8 10 3

8 10 5 5 5 7 9 10 2 6 1 5 8 6

9 5 9 3 10 3 6 1 8 9 10 5 9 3

Output 01

22

Copy

13 8 11 7 9 6 7 5 5 4 3 5 4 3

13 8 11 7 9 6 7 5 5 6 3 5 4 3

13 8 11 7 9 6 7 7 5 6 3 5 4 3

[Quay Lui - Nhánh Cận]. Bài 34. Đường đi dài nhất

Submit

Time limit: 1.0 / **Memory limit:** 256M

Point: 100

Cho một mảng 2 chiều **A[][]** kích thước **N x M**. Mỗi ô trên mảng 2 chiều là số 1 tương ứng với đường đi hoặc 0 tương ứng với vật cản, nhiệm vụ của bạn là hãy tìm đường đi dài nhất (có số ô đi qua dài nhất) từ ô **(s, t)** tới ô **(u, v)** hoặc kết luận rằng không thể đi tới ô **(u, v)**.

Mỗi lần di chuyển bạn được phép đi theo 4 hướng là **trên (U)**, **dưới (D)**, **phải (R)** và **trái (L)**.



đường đi mà nó tự nó chọn càng dài...

Đầu vào

- Dòng 1 bắt đầu bởi các số nguyên **N**, **M** và **s**, **t**, **u**, **v**.
- **N** dòng tiếp theo, mỗi dòng gồm **M** số nguyên **A[i][j]**

Giới hạn

- $1 \leq N, M \leq 15$
- $0 \leq A[i][j] \leq 10$

Đầu ra

- Dòng 1 in ra độ dài đường đi dài nhất tìm được.
- Các dòng tiếp theo mỗi dòng là được đi qua từng ô của con Mã, coi đường đi là một chuỗi ký tự, hãy in các đường đi theo thứ tự từ điển tăng dần. Nếu không có đường đi hãy in ra -1.

Ví dụ :

Input 01

```
7 6
5 6 7 1
0 1 0 1 0 1
1 1 1 0 1 0
1 1 1 1 1 1
1 1 1 0 1 0
1 1 0 1 1 1
1 0 1 0 1 0
1 1 0 1 0 1
```

[Copy](#)

Output 01



```
LUULLDLUULDDDDD
LUULLLULDDRDLDD
LUULLULDLDRDLDD
LUULLULLDDRDLDD
LUULLULLDRDDLDD
LUULLULLDRDLDDD
```

Input 02

```
6 4
3 2 2 1
1 1 1 1
1 0 1 0
0 1 0 1
1 0 1 1
1 1 0 1
1 1 1 0
```

[Copy](#)

Output 02

```
-1
```

[Copy](#)

Được phát triển bởi **28TECH.COM.VN**