**Problem: Add two integers**

**Description**

Compute the sum of two integers a and b.

**Input**

* Line 1 contains two integers a and b (0 <= a, b <= 1019)

**Ouput**

Write the sum of a and b

**Example**

**Input**

3 5

**Output**

8

**Problem: Longest Path on a Tree**

**Description**

Given a undirected tree $G = (V, E)$ in which $V = \{1,…,N\}$ is the set of nodes. Each edge $(u,v) \in E$ has weight $w(u,v)$. The length of a path is defined to be the sum of weights of edges of this path. Find the longest elementary path on $G$.

**Input**

* Line 1: positive integer $N$ ($1 \leq N \leq 10^5$)
* Line $i + 1$ ($i = 1,…,N-1$): positive integers $u, v, w$ in which $w$ is the weight of edge $(u,v)$ (1 ≤ w ≤ 100)

**Output**

The weight of the longest path on the given tree

**Example**

**Input**

6

1 3 3

1 6 2

2 6 5

4 5 2

4 6 1

**Output**

**10**

**Problem: Compute Strongly Connected Components of directed graphs**

**Description**

Given a directed graph G=(V,E) where V={1,. . ., N} is the number of nodes and the set E has M arcs. Compute number of strongly connected components of G

**Input**

* Line 1: two positive integers N and M (1 <= N <= 105, 1 <= M <= 106)
* Line i+1 (i=1,. . ., M\): contains two positive integers u and v which are endpoints of ith arc

**Output**

Write the number of strongly connected components of G

**Example**

**Input**

8 13 1 2 1 8 2 3 2 6 3 6 4 3 4 6 5 4 6 5 7 1 7 2 7 6 8 7

**Output**

3

**Problem: Compute C\_k\_n (Dynamic Programing) (recursion with memoization)**

**Description**

Given two positive integers k and n. Compute C(k,n) which is the number of ways to select k objects from a given set of n objects.

**Input**

* Line 1: two positive integers k and n (1 <= k,n <= 999)

**Output**

Write te value C(k,n) modulo 109+7.

**Example**

**Input**

3 5

**Output**

10

**Problem: Binary sequence generation (Back Tracking)**

**Description**

Given an integer n, write a program that generates all the binary sequences of length n in a lexicographic order.

**Input**

* Line 1: contains an integer n (1 <= n <= 20)

**Output**

Write binary sequences in a lexicographic ordder, eac sequence in a line

**Example**

**Input**

3

**Output**

000

001

010

011

100

101

110

111

**Problem: List order of nodes visited by a DFS**

**Description**

Given a undirected graph =(V,E) in which V = {1,2,..,n} is the set of nodes. Write a program that visit nodes of G by a DFS (consider a lexicorgraphic order of nodes).

**Input**

* Line 1: contains 2 integers n and m (1 <= n,m <= 100000)
* Line i+1: contains u and v which are two end-points of the ith edge

**Output**

Sequence of nodes visited by DFS

**Example**

**Input**

7 12

1 2

1 3

2 3

2 4

2 7

3 5

3 7

4 5

4 6

4 7

5 6

5 7

**Output**

1 2 3 5 4 6 7

**Problem: Max SubSequence**

**Description**

Given an integers sequence a=(a1, a2,…, an). A subsequence of aa is defined to be ai, ai+1,…, aj. The weight of a subsequence is the sum of its elements. Find the subsequence having the highest weight.

**Input**

* Line 1 contains n (1≤n≤106)
* Line 2 contains a1,…,an (−500≤ai≤500)

**Output**

Write the weight of the highest subsequence found.

**Example**

**input**

10 3 -1 -3 5 2 5 0 -1 5 4

**output**

20

**Problem: Count solutions to an integer linear equation**

**Description**

Given a positive integer n and n positive integers a1, a2, ..., an. Compute the number of positive integer solutions to the equation:

a1X1 + a2X2 + . . . + anXn = M

**Input**

* Dòng 1: n và M
* Dòng 2: a1, a2, ..., an

**Output**

Số nghiệm nguyên dương

**Ví dụ**

**Input**

3 5

1 1 1

**Output**

6

**Problem: Max Independent Set on a Tree**

**Description**

Cho cây T = (V,E) trong đó V = {1,…,n} là tập các đỉnh và E là tập gồm n-1 cạnh. Mỗi đỉnh v của đồ thị có trọng số w(v). Hãy chọn ra tập S các đỉnh sao cho hai đỉnh kề nhau không cùng được chọn và tổng trọng số các đỉnh được chọn là lớn nhất.

**Dữ liệu**

· Dòng 1: ghi 2 số nguyên dương n (1 <= n <= 20, 1 <= m <= 100)

· Dòng 2: ghi n số nguyên dương w(1), w(2), …, w(n) là trọng số các đỉnh tương ứng

· Dòng i+2 (i = 1,…,n-1): ghi 2 số nguyên dương u và v là 2 đầu mút của cạnh thứ i của T

**Kết quả:**

· Ghi ra tổng trọng số của các đỉnh được chọn

Ví dụ

**Dữ liệu**

9

9 5 10 4 3 2 6 1 8

1 5

2 3

2 4

2 6

3 5

6 7

6 8

6 9

**Kết quả**

38

**Problem: BACP**

**Description**

The BACP is to design a balanced academic curriculum by assigning periods to courses in a way that the academic load of each period is balanced. There are N courses 1, 2, …, N that must be assigned to M periods 1, 2, …, M. Each course i has credit ci and has some courses as prerequisites. The load of a period is defined to be the sum of credits of courses assigned to that period. The prerequisites information is represented by a matrix ANxN in which Ai,j = 1 indicates that course i must be assigned to a period before the period to which the course j is assigned. Compute the solution satisfying constraints:

* Satisfy the prerequisites constraints: if Ai,j = 1, then course i must be assigned to a period before the period to which the course j is assigned
* The maximum load for all periods is minimal

**Input**

* Line 1 contains N and M (2 ≤ N ≤16, 2 ≤ M ≤ 5)
* Line 2 contains c1, c2, …, cN
* Line i+2 (i = 1,…, N) contains the ith line of the matrix A

**Output**

* Unique line contains that maximum load for all periods of the solution found

**Example**

**Input**

6 2

4 4 4 4 2 4

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 1 0 0 0

0 0 1 0 0 0

1 0 0 0 0 0

**Output**

12

**Problem: WATER JUGS**

**Description**

There are two jugs, a-litres jug and b-litres jug (a, b are positive integers). There is a pump with unlimited water. Given a positive integer c, how to get exactly c litres.

**Input**

Line 1: contains positive integers a, b, c (1 <= a, b, c <= 900)

**Output**

write the number of steps or write -1 (if no solution found)

**Example**

**Input**

6 8 4

**Output**

4

**Problem: Generate Keys in DBMS**

**Description**

Trong hệ quản trị cơ sở dữ liệu (DB) đã lưu trữ các đối tượng có khóa (key) là các xâu ký tự độ dài đều bằng L và mỗi ký tự đều là các con số từ 0 đến 9. Khóa của các đối tượng không được phép trùng lặp.

Hiện tại, DB đã có các khóa là k1, k2, ..., kn. Người dùng muốn thêm m đối tượng nữa vào cơ sở dữ liệu. DB phải sinh ra m khóa nhỏ nhất khác nhau nữa có độ dài bằng L và không trùng với n khóa đã có.

Lưu ý: DB không chấp nhận khóa gồm toàn chữ số 0.

Hãy viết chương trình giúp DB sinh ra m khóa mới này.

**Dữ liệu**

* Dòng 1: n, L, m (1 <= n,m <= 10000, 1 <= L <= 32)
* Dòng i+1 (i=1,...,n): ghi khóa ki

**Kết quả**

Dòng i (i = 1,2,...,n+m) ghi khóa thứ i (sắp xếp theo thứ tự từ điển) trong dãy khóa bao gồm cả khóa đã có trong DB và khóa mới

**Ví dụ**

**Dữ liệu**

4 6 10

000011

000002

000004

000003

**Kết quả**

000001

000002

000003

000004

000005

000006

000007

000008

000009

000010

000011

000012

000013

000014

**Problem: Text Replacement**

**Description**

Cho văn bản T và 2 mẫu P1, P2 đều là các xâu ký tự (không chứa ký tự xuống dòng, độ dài không vượt quá 1000). Hãy thay thế các xâu P1 trong T bằng xâu P2.

*Dữ liệu*

· Dòng 1: xâu P1

· Dòng 2: xâu P2

· Dòng 3: văn bản T

*Kết quả*:

· Ghi văn bản T sau khi thay thế

Ví dụ

*Dữ liệu*

AI

Artificial Intelligence

Recently, AI is a key technology. AI enable efficient operations in many fields.

*Kết quả*

Recently, Artificial Intelligence is a key technology. Artificial Intelligence enable efficient operations in many fields.

**Problem: All pair shortest paths**

**Description**

Given a directed graph G = (V, E) in which V = {1, 2, ..., n} is the set of nodes, and w(u,v) is the weight (length) of the arc(u,v). Compute d(u,v) - the length of the shortest path from u to v in G, for all u,v in V.

Input

Output

**Problem: Hamiton Cycle**

**Description**

Given an undirected graph G = (V,E). Write a program to check if G is a Hamiltonian graph.

**Input**

* Line 1: a positive integer T (number of graphs)
* Subsequent lines are information about T graphs, each has the following format:
* Line 1: n and m (number of nodes and edges)
* Line i+1 (i = 1, 2, ..., m): u and v : two end points of the ith edge

**Output**

* In the ith line, write 1 if the corresponding is a Hamiltonian graph, and write 0, otherwise

**Example**

**Input**

2

5 5

1 2

1 3

2 4

2 5

3 5

7 13

1 3

1 5

1 7

2 4

2 5

2 6

3 4

3 5

3 7

4 6

4 7

5 7

6 7

**Output**

0

1

**Problem: Job Selection & Planning (GREEDY)**

**Description**

Given n jobs J = {1,. . .,n}. Each job i has a deadline d(i) and associated profit p(i) if the job is finished before the deadline. Every job takes the single unit of time, so the minimum possible deadline for any job is 1. It is also noted that no more than one job can be executed at a time.

Select and schedule a subset of jobs of J such that the total profits is maximal.

**Input**

* Line 1: contains a positive integer n (1 <= n <= 105)
* Line i+1 (i=1,. . . ,n) contains d(i) and p(i) (1 <= d(i), p(i) <= 105)

**Output**

* Write to total profits obtained from the subset of jobs found.

**Example**

**Input**

6

3 10

2 40

6 70

3 50

5 80

1 60

**Output**

300

**Problem: Longest Common Subsequence**

**Description**

Denote *X* = *X*1, *X*2, …, *Xn*, a subsequence of *X* is created by removing some element from *X.* Given two sequences *X* = *X*1, *X*2, …, *Xn* and *Y* = *Y*1, *Y*2, …, *Ym.* Find a common subsequence of *X* and *Y* such that the length is the longest.

**Input**

* Line 1: two positive integers n and m (1 <= n,m <= 10000)
* Line 2: n integers *X*1, *X*2, …, *Xn*
* Line 3: m integers *Y*1, *Y*2, …, *Ym*

**Output**

* Length of the longest subsequence of X and Y

**Example**

**Input**

7 10

3 7 2 5 1 4 9

4 3 2 3 6 1 5 4 9 7

**Output**

5

**Problem: Capacitated Vehicle Routing - Not Empty Route Allowed**

**Description**

A fleet of K identical trucks having capacity Q need to be scheduled to delivery pepsi packages from a central depot 0 to clients 1,2,…,n. Each client i requests d[i] packages. The distance from location i to location j is c[i,j], 0≤i,j≤n. A delivery solution is a set of routes: each truck is associated with a route, starting from depot, visiting some clients and returning to the depot for deliverying requested pepsi packages such that:

* Each client is visited exactly by one route
* Total number of packages requested by clients of each truck cannot exceed its capacity
* Each truck must visit at least one client

Goal

* Find a solution having minimal total travel distance

Note that: the orders of clients in a route is important, e.g., routes 0 -> 1 -> 2 -> 3 -> 0 and 0 -> 3-> 2 -> 1 -> 0 are different.

**Input**

* Line 1: n,K,Q (2≤n≤12,1≤K≤5,1≤Q≤50)
* Line 2: d[1],...,d[n](1≤d[i]≤10)
* Line i+3 (i=0,…,n): the ith row of the distance matrix c (1≤c[i,j]≤30)

**Output**

Minimal total travel distance

**Example**

**Input**

4 2 15

7 7 11 2

0 12 12 11 14

14 0 11 14 14

14 10 0 11 12

10 14 12 0 13

10 13 14 11 0

**Output**

70

**Problem: Capacitaed Vehicle Routing**

**Description**

A fleet of K identical trucks having capacity Q need to be scheduled to delivery pepsi packages from a central depot 0 to clients 1,2,…,n. Each client i requests d[i] packages. The distance from location i to location j is c[i,j], 0≤i,j≤n. A delivery solution is a set of routes: each truck is associated with a route, starting from depot, visiting some clients and returning to the depot for deliverying requested pepsi packages such that:

* Each client is visited exactly by one route
* Total number of packages requested by clients of each truck cannot exceed its capacity

Goal

* Find a solution having minimal total travel distance

Note that:

* There might be the case that a truck does not visit any client (empty route)
* The orders of clients in a route is important, e.g., routes 0 -> 1 -> 2 -> 3 -> 0 and 0 -> 3-> 2 -> 1 -> 0 are different.

**Input**

* Line 1: n,K,Q (2≤n≤12,1≤K≤5,1≤Q≤50)
* Line 2: d[1],...,d[n](1≤d[i]≤10)
* Line i+3 (i=0,…,n): the ith row of the distance matrix c (1≤c[i,j]≤30)

**Output**

Minimal total travel distance

**Example**

**Input**

4 2 15

7 7 11 2

0 12 12 11 14

14 0 11 14 14

14 10 0 11 12

10 14 12 0 13

10 13 14 11 0

**Output**

70

**Problem: Paper Reviewer Assignment Problem**

**Description**

The chair of a conference must assign scientific papers to reviewers in a balance way. There are N papers 1, 2, …, N and M reviewers 1, 2, …, M. Each paper i has a list L(i) of reviewers who are willing to review that paper. A review plan is an assignment reviewers to papers. The load of a reviewer is the number of papers he/she have to review. Given a constant b, compute the assignment such that

* Each paper is reviewed by exactly b reviewers • The maximum load of all reviewers is minimal

Input

* Line 1 contains N, M and b
* Line i+1 (i = 1,…,N) contains a positive integer k followed by k positive integers representing the list L(i)

Output

* Unique line contains the maximum load for all reviewers of the solution found or contains -1 if no solution found

**Problem: Representation N as the sum of positive integers**

**Description**

Given a positive integer N. Compute the number Q of ways to represent N as the sum of positive integers.

**Input**

* Line 1: contains a positive integer N (1 <= N <= 1000)

**Output**

* Write the value Q modulo 109+7

**Example**

**Input**

4

**Output**

5

**Problem: Max Even SubSequence**

**Description**

Given a sequence of n integers a=a1, . . ., an. A subsequence of a consists of contiguous elements of a (for example, ai, ai+1, . . . ,aj). The weight of a subsequence is defined to be the sum of its elements. A subsequence is called even-subsequnce if its weight is even. Find the even-subsequence of a having largest weight.

**Input**

* Line 1: contains a positive integer n (1 <= n <= 106)
* Line 2: contains a1, . . ., an (-106 <= ai <= 106)

**Output**

* The weight of the largest even-subsequence found, or write NOT\_FOUND if no solution found.

**Example**

**Input**

4

1 2 -3 4

**Output**

4

**Problem: Count Spanning Tree**

**Description**

Given a undirected connected graph G = (V,E) in which V = {1,…,N} is the set of nodes and E is the set of M edges. Count the number of spanning trees of G.

**Input**

* Line 1: contains positive integers N and M (1 <= N <= 20, 1 <= M <= 25)
* Line i+1 (i = 1,…, M): contains u and v which are endpoints of the ith

**Output**

* Write the number of spanning trees of G

Example

Input

Output

**Problem: Sum of length of paths from root on a tree (DFS)**

**Description**

Given a tree T=(V,E) in which V={1,. . ., n} is the set of nodes. Each edge (u,v) of T has length w(u,v). Denote f(v) the sum of length of paths from all other nodes to v. Write a program to compute max{f(1), . . ., f(n)}.

Input

* Line 1 contains a positive integer n (2 <= n <= 10^5)
* Line i+1 (i=1, . . ., n): contains u, v and w in which w is the weight of the edge (u,v)

Output

* Write the value max{f(1), . . , f(n}.

Example

Input

Output

**Problem: Count number of sudoku solutions (back tracking)**

**Description**

Write a program to compute the number of sudoku solutions (fill the zero elements of a given partial sudoku table)

Fill numbers from 1, 2, 3, .., 9 to 9 x 9 table so that:

* Numbers of each row are distinct
* Numbers of each column are distinct
* Numbers on each sub-square 3 x 3 are distinct

**Input**

* Each line *i* (*i* = 1, 2, ..., 9) contains elements of the *i*th row of the Sudoku table: elements are numbers from 0 to 9 (value 0 means the empty cell of the table)

**Output**

* Write the number of solutions found

**Example**

**Input**

0 0 3 4 0 0 0 8 9

0 0 6 7 8 9 0 2 3

0 8 0 0 2 3 4 5 6

0 0 4 0 6 5 0 9 7

0 6 0 0 9 0 0 1 4

0 0 7 2 0 4 3 6 5

0 3 0 6 0 2 0 7 8

0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0

**Output**

64

**Problem: Gold Mining (Dynamic programming, deque)**

**Description**

The Kingdom ALPHA has n warehouses of golds located on a straight line and are numbered 1, 2,..., n. The warehouse i has amount of ai (aiai is non-negative integer) and is located at coordinate i (i=1,...,n). The King of ALPHA opens a competition for hunters who are responsible to find a subset of gold warehouses having largest total amount of golds with respect to the condition that the distance between two selected warehouses must be greater than or equal to L1 and less than or equal to L2.

**Input**

* Line 1 contains n, L1, and L2 (1≤n≤1000000,1≤L1≤L2≤n)
* Line 2 contains n integers a1,a2,…,an

**Output**

Contains only one single integer denoting the total amount of golds of selected warehouses.

**Example**

**Input**

6 2 3

3 5 9 6 7 4

**Output**

19

**Problem: Bridges and Articulation Points**

**Description**

Given an undirected graph containing N vertices and M edges, find all the articulation points and the bridges in the graph.

**Input**

* The first line consists of two space-separated integers denoting N and M,
* M lines follow, each containing two space-separated integers X and Y denoting there is an edge between X and Y.

**Output**

* One line consists of two integers denoting the number of articulation points and the number of bridges.

**Example**

**Input**

10 12

1 10

10 2

10 3

2 4

4 5

5 2

3 6

6 7

7 3

7 8

8 9

9 7

**Output**

4 3

**Problem: Max Flow**

**Description**

Given a network G = (V, E) which is a directed weighted graph. Node s is the source and node t is the target. c(u,v) is the capacity of the arc (u,v). Find the maximum flow on G.

**Input**

•Line 1: two positive integers N and M (1 <= N <= 104, 1 <= M <= 106)

•Line 2: contains 2 positive integers s and t

•Line i+2 (I = 1,. . ., M): contains two positive integers u and v which are endpoints of ith arc

**Output**

Write the value of the max-flow found

**Example**

**Input**

7 12

6 7

1 7 7

2 3 6

2 5 6

3 1 6

3 7 11

4 1 7

4 2 4

4 5 5

5 1 4

5 3 4

6 2 8

6 4 10

**Output**

17

**Problem: Permutation generation (back tracking)**

**Description**

Given an integer n, write a program to generate all permutations of 1, 2, ..., n in a lexicalgraphic order (elements of a permutation are separated by a SPACE character).

**Example**

**Input**

3

**Output**

1 2 3

1 3 2

2 1 3

2 3 1

3 1 2

3 2 1