

G. Harvester

time limit per test: 0.5 seconds
 memory limit per test: 256 megabytes
 input: standard input
 output: standard output

It is Bubble Cup finals season and farmer Johnny Bubbles must harvest his bubbles. The bubbles are in a rectangular bubblefield formed of $N \times M$ square parcels divided into N rows and M columns. The parcel in i^{th} row and j^{th} column yields $A_{i,j}$ bubbles.

Johnny Bubbles has available a very special self-driving bubble harvester that, once manually positioned at the beginning of a row or column, automatically harvests all the bubbles in that row or column. Once the harvester reaches the end of the row or column it stops and must be repositioned. The harvester can pass through any parcel any number of times, but it can collect bubbles from the parcel only once.

Johnny is very busy farmer, so he is available to manually position the harvester at most four times per day. Johnny is also impatient, so he wants to harvest as many bubbles as possible on the first day.

Please help Johnny to calculate what is the maximum number of bubbles he can collect on the first day.

Input

The first line contains two integers N and M ($1 \leq N, M \leq N * M \leq 10^5$) - the bubblefield size.

Each of the next N lines contains M integers. The j^{th} element in the i^{th} line is $A_{i,j}$ ($0 \leq a_{i,j} \leq 10^9$) — the yield of the parcel located in the i^{th} row and the j^{th} column.

Output

Output contains one integer number - maximum number of the bubbles Johnny can harvest on the first day.

Examples

input	Copy
2 2 1 2 3 4	
output	Copy
10	

input	Copy
5 5 0 9 2 7 0 9 0 3 0 5 0 8 0 3 1 6 7 4 3 9 3 6 4 1 0	
output	Copy
80	

Note

In the first example, farmer Johnny can harvest all the bubbles by positioning the harvester on the first and the second row.

In the second example, one way Johnny can harvest maximum number of bubbles is to position the harvester in the second row, the fourth row, the second column and the fourth column.

Bubble Cup 12 - Finals [Online Mirror, unrated, Div. 2]

[Finished](#)
[Practice](#)


→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

[Start virtual contest](#)

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup

You can clone this contest to a mashup.

[Clone Contest](#)

→ Submit?

Language: GNU G++11 5.1.0

Choose file: [选择文件](#) [未选择任何文件](#)




[Submit](#)

→ Problem tags

[implementation](#) [*2000](#)

No tag edit access

→ Contest materials

- Announcement (en) 
- Statements (en) 
- Tutorial (en) 



[Codeforces](#) (c) Copyright 2010-2019 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Sep/19/2019 06:57:35^{UTC+8} (e1).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#)

Supported by



ITMO UNIVERSITY