

Coding Area

 tcscodevita.com/main_page.jsp

05Hr 42Min 00Sec

Your Contest Ends At
2022-03-12 01:19:46 IST

- Guidelines
- Coding Area
- Public Testcase Submissions
- Private Testcase Submissions
- Unevaluated Submissions
- Feedback Form
- Graphs

Online Editor (D)

A

B

C

D

E

F

Docstore Offline



Problem Description

Docstore is a utility which is used to store documents in a file/container. And while writing or storing the file you must know the exact location of the document where you are writing it. You must know what is the starting offset and ending offset of that particular file. And these all entries are inserted into the database so that at the time of reading of file you directly start reading from that offset and return the file.

You have been provided a list of files and its subsequent details. Soft deletion flag will be given. You can use that space to accommodate the new file. If deletion flag is 0 file is not soft deleted. You have been given a list of files with their size. You have insert file at a place where it gets the best fit.

Best fit- Order will be

- Exact match with the free space.
- Insert where there is Least Internal fragmentation.
- If none of them is possible append after last file.

Print the starting offset of every file inserted in sorted order of file name.

Note- b=byte, kb=kilobytes, mb=megabyte

+

Constraints

$1 < N < 100$

$1 < M < 100$

+

Input

First line contains an integer denoting the number of files (N) already present in the disk.

Next lines contain space separated data about every file in the format given below

Filename Offset Size DeletionStatus

Next line contains an integer denoting the number of files(M) to be inserted.

Next M lines contains space separated data about every file in the format given below

Filename Size

+

Output

Print the starting offset of every file inserted in sorted(lexicographically) order of file name.

+

Time Limit (secs)

1

+

Examples

Input

4

File1 0 2b 0

File2 2 3b 1

File3 5 1b 0

File4 6 4b 0

2

File5 2b

File6 3b

Output

File5 2

File6 10

Explanation-

Here we have to add file5 so will check of free spaces with exact match. But there is no exact match. Now we will check for least internal fragmentation. So file2 is deleted and we have 3b space. So, we can accommodate file5 at the position of file2 with 1b internal fragmentation.

Now we have to insert file6 so will check of free spaces with exact match. But there is no exact match. Now we will check for least internal fragmentation. But here we don't have any space which can accommodate the file. Finally, we will add that file after the last file written. Hence it will be stored at offset 10 after file4 completion.

Example 2

Input

3

File1 0 2b 0

File2 2 3b 0

File3 5 1b 0

2

File4 1b

File5 7b

Output

File4 6

File5 7

Explanation-

Here we have to add file4 and file5 so will check of free spaces and all the cases but no space is present to occupy any file so they will be added at the last. One after another.

Upload Solution [Question : D]

Warning !

Right click is disabled.