



G - Filesystem

Given a series of file system commands and queries, output the contents of the file requested by each query (a query is a filename to be printed out). If the query is invalid (e.g. path does not exist), the text `invalid!` must be printed.

The following commands must be supported:

- `echo "<content>" > <path>`

Writes `<content>` to the file `<path>`. If `<path>` exists, it must be a file, and the contents will be replaced. If the path does not exist, its parent directory must exist, and the file will be created. `<content>` will only contain ASCII numbers, letters, `*`, `?`, `<`, `>`, and space.

- `cp <source> <destination>`

Copies file `<source>` to file or directory `<destination>`. If `<destination>` already exists and is a directory, the file will be copied to the `<destination>` directory, with a filename which is the final path component of `<source>`. Otherwise, `<destination>` is the new filename, and its parent directory must already exist. The new file replaces any existing content.

- `mv <source> <destination>`

Moves file or directory `<source>` to the file or directory `<destination>`. Similarly to `cp`, if `<destination>` already exists and is a directory, the source file/directory will be moved to a path within the `<destination>` directory whose name matches the final path component of `<source>`. Otherwise, `<destination>` is the new filename, and its parent directory must already exist. The new file replaces any existing content, and `<source>` is removed.

- `rm <path>`

Deletes the file with the path `<path>`.

- `mkdir <path>`

Creates a directory with the path `<path>`. The parent directory must already exist.

- `rmdir <path>`

Removes an empty directory with the path `<path>`.

All filesystem paths will only contain numbers, letters, or `/` and will not exceed 31 characters or 7 path components. Paths components will be separated by a `/`. All commands in the input are valid and will not refer to non-existent/invalid paths unless behavior is specified above. All commands will be less than 1024 characters.

Input

The first line of input contains T ($0 < T < 100$), indicating the number of test cases.

The next line of input contains C ($0 < C < 100$), indicating the number of commands for the test case. The next C lines will contain commands to generate the filesystem. The folder that the filesystem starts in may be ignored. In other words, the current directory ($./$) may be assumed.

The next line of input contains Q ($0 < Q < 20$), indicating the number of filenames for the test case whose contents should be printed. The next Q lines will contain filenames (or a path & filename, e.g. `myfolder/myfile`) to print out the contents of a given file.

Output

Output the contents of the file a query requests or "invalid!" if the file does not exist.

Print out a blank after each case.

Sample Input

```
2
2
echo "This is foo" > foo
cp foo bar
2
foo
bar
3
mkdir dir
echo "Bar" > bar
mv bar dir/bar
1
bar
```

Sample Output

```
This is foo
This is foo

invalid!
```