

Shmoogle Wave

Shmoogle company developed new protocol <<Wave>> for the multi-user online editing of the text data. Using it, any client can send commands to the server for editing the text and the server would broadcast it to all other connected clients. Each command consists of a sequence of operations of the following type:

Operation	Description
R k	Moves the cursor k positions right. The cursor is positioned at before the first character of the text at the start of performing the command.
C k s	Inserts the string s of length k at the cursor position. After this operation the cursor is positioned to the right of the inserted string.
D k	Deletes k characters right of the cursor.

When the new client connects the server needs to send it all the command this client missed. In order to save traffic the server merges all the commands in one equivalent command. Help Shmoogle company implement merging of commands effectively.

The resulting command should consist of the least possible number of operations. The delete operations should precede the insert operations if possible.

Input

The first line contains T ($1 \leq T \leq 10$) — the number of test cases. The description of T tests follow. The first line of each test case contains the amount of commands n ($1 \leq n \leq 10000$). The description of each command follows. The first line of each command contains the amount of operations m ($1 \leq m \leq 10$). The next m lines contain the description of each operation in the format given above. $1 \leq k \leq 100000$ for R and D operations, and $1 \leq k \leq 10$ for C operations. The strings in C operations consist of latin letters and digits only.

Output

For each test case your program should print the result of merging the commands. The format of the command should be the same as in the input file, except for the limitations on m and k . The result should consist of the least possible number of operations. The delete operations should precede the insert operations if possible. If the result of merging consist of no operations print 0.

Example

Input:

```
1
2
4
R 4
C 3 abc
R 2
C 3 xyz
3
R 7
```

C 3 def
D 3

Output:

3
R 4
D 2
C 8 abcdefyz