

Нека имаме сграда с N етажа и в нея има асансьор, който обслужва заявките в реда, в който са постъпили (настинат е бутон). Скоростта на асансьора е константна и е 0.2 етажа в секунда (т.е минава един етаж за 5 секунди). Времето за престой на етаж приемете за 0.

Целта на вашата програма е да реализира симулация на работата на асансьора при дадени заявки за към него.

Входните данни ще се четат от текстов файл, името на който ще получите като аргумент от командния ред на програмата си (през `argv`). Първият ред на входа съдържа 2 цели числа N и K . N е броят етажи в сградата, а K е броят заявки, които следват, като всяка е на отделен ред. Заявките за асансьора са 2 типа:

- Повикване от етаж: `call DIR FLOOR TIME`
 - `DIR`: посока, едно от {up, down}
 - `FLOOR`: етаж, цяло число в интервала $[1, N]$
 - `TIME`: време в секунди от началото на симулацията
- Команда от вътре: `go FLOOR TIME`
 - `FLOOR`: етаж, цяло число в интервала $[1, N]$
 - `TIME`: време в секунди от началото на симулацията

За изход на вашата програма трябва да изведете `TIME`, `FLOOR`, `DIR` за всяко спиране на асансьора на етаж, където `TIME` са секундите изминали от началото на симулацията, `FLOOR` е етаж, на който е спрял асансьорът, а `DIR` е посоката, от която е дошъл на този етаж.

Забележка 1:

Секундите в заявките нямат общо с времето за изпълнение на вашата програма. Заявките може да приемете като списък от вече приключили събития. В момент 0 асансьорът е на етаж 1.

Забележка 2:

Асансьорът обработва заявките по ред на тяхното постъпване. Единственото изключение на това правило, е когато той може да спре на някой етаж без да променя текущо изпълняваната заявка. Например ако се движи от етаж 1 до етаж 5 и получи заявка за етаж 4, докато се намира на етаж 2, то той ще спре на етаж 4.