



Упражнения: Допълнителни задачи за абстракция и интерфейси

1. Граничен контрол

Бъдещето е! Вие сте владетел на тоталитарно деспотично общество населено с граждани и роботи и понеже се страхувате от безредици, решавате да имплементирате сериозен контрол върху това кой влиза в града ви. Вашите войници проверяват Id-тата на всеки, който влиза и излиза.

Ще получите неизвестно количество редове от конзолата до получаване на командата "End". На всеки ред ще има информация за гражданин или робот, който се опитва да влезе в града във формат "<name> <age> <id>" за граждани и "<model> <id>" за роботи. След командата за край, на следващия ред ще получите номер, който показва на колко завършват фалшивите Id номера, всички граждани или роботи с фалшиви Id трябва да бъдат арестувани.

Изходът от програмата трябва да съдържа всички Id-та на арестуваните, като всяко е на отделен ред.

Вход

Входът идва от конзолата. Параметрите на всяка команда ще бъдат разделени с по един интервал.

Примери

Вход	Изход
Pesho 22 9010101122 MK-13 558833251 MK-12 33283122 End 122	9010101122 33283122
Toncho 31 7801211340 Penka 29 8007181534 IV-228 999999 Stamat 54 3401018380 KKK-666 80808080 End 340	7801211340



2. Рожден ден

Известен факт е, че хората празнуват рождениците си дни. Известно е също, че някои хора празнуват и рождениците дни дори на своите домашни любимци. Модифицирайте програмата от предната задача, така че да добавите дати на раждане към гражданите си и добавете клас Pet, домашните любимци имат име и дата на раждане. Разпределете повтарящата се функционалност в интерфейси и ги имплементирайте в класовете си.

Ще получите неизвестно количество редове от конзолата до получаване на командата "End". На всеки ред ще има информация във формат "Citizen <name> <age> <id> <birthdate>" за граждани, "Robot <model> <id>" за роботи или "Pet <name> <birthdate>" за домашни любимци. След командата за край, на следващия ред ще получите номер, който е конкретна година, като вашата задача е да отпечатате всички рождени дати от тази година на всички граждани и домашни любимци във формат ден/месец/година.

Примери

Вход	Изход
Citizen Pesho 22 9010101122 10/10/1990 Pet Sharo 13/11/2005 Robot MK-13 558833251 End 1990	10/10/1990
Citizen Stamat 16 0041018380 01/01/2000 Robot MK-10 12345678 Robot PP-09 00000001 Pet Topcho 24/12/2000 Pet Kosmat 12/06/2002 End 2000	01/01/2000 24/12/2000
Robot VV-XYZ 11213141 Citizen Penka 35 7903210713 21/03/1979 Citizen Kane 40 7409073566 07/09/1974 End 1975	<empty output>



3. Недостиг на храна

Вашето тоталитарно геспотично общество страда от недостиг на храна, затова се появяват много бунтовници. Разширете кода от предната задача с нова функционалност, за да решите тази.

Дефинирайте клас Rebel с поле, age и group (низ), имената са уникални – няма да има 2 бунтовници/граждани или бунтовник и гражданин с едно и също име. Дефинирайте интерфейс IBuyer, който дефинира метод BuyFood() и свойство Food, което да е цяло число. Имплементирайте IBuyer интерфейса в Citizen и Rebel класа, като и бунтовниците и гражданите започват с 0 храна, когато бунтовник купи храна, неговата стойност за Food се увеличава с 5, когато гражданин купи храна, неговата стойност за Food се увеличава с 10.

На първия ред от входа ще получите цяло число N – броят на хората, на всеки от следващите N реда ще получите информация в един от следните формати: "<name> <age> <id> <birthdate>" за гражданин или "<name> <age><group>" за бунтовник. След N реда ще получавате имена на хора, които са купили храна, като всяко име ще е на отделен ред. Ще получавате имена, докато не получите команда "End". Забележете, че не всички имена може да са валидни. В случай, че името е невалидно – трябва да го игнорирате.

Изход

Изходът се състои от само един ред, на който трябва да отпечатате общото количество закупена храна.

Примери

Вход	Изход
2 Pesho 25 8904041303 04/04/1989 Stancho 27 WildMonkeys Pesho Gosho Pesho End	20
4 Stamat 23 TheSwarm Toncho 44 7308185527 18/08/1973 Joro 31 Terrorists Penka 27 881222212 22/12/1988 Jiraf Joro Jiraf Joro Stamat Penka End	25