

Игра

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Это интерактивная задача.

Крош пришёл в очередной раз поиграть в компьютер Лосяша, но вместо привычных игр «Принц для Юши 2» или «Крутой спуск» он обнаружил новую игру «Коварные демонюги». Крошу очень понравилась эта игра, но он постоянно проигрывал. Помогите Крошу составить стратегию, с которой он будет выигрывать чаще, чем проигрывать.

В данной задаче вам будут предоставлены правила игры против интерактора в «Коварные демонюги». Чтобы решить эту задачу, вам нужно выиграть у интерактора хотя бы 500 из 1000 игр. Интерактор **не является** адаптивным, то есть не подстраивается под ваш стиль игры.

«Коварные демонюги» — карточная игра, в которой карты пронумерованы и выложены по кругу. В этой игре у каждой карты есть своя роль.

Представим роли:

1. **Рыцарь**. Это персонаж, которого нельзя *убить*, если он не *врёт*.
2. **Охотник**. Это персонаж, который скажет, на каком расстоянии от него находится **Демон**, но не скажет, в каком направлении. Если этот персонаж врёт, он скажет случайное число от 1 до $\frac{n}{2}$ с округлением вниз.
3. **Гренадер**. Если вы убьёте этого персонажа, когда он не врёт, игра заканчивается поражением.
4. **Шут**. Персонаж, который позволяет выбрать три различные карты, и он скажет, сколько **Демонов** среди них. Способность **Шута** можно использовать только один раз за игру. Если будут выбраны не различные карты, игра закончится поражением.
5. **Судья**. Персонаж, который позволяет выбрать одну карту и узнать, *врёт* ли она. Эту способность можно использовать только один раз за игру.
6. **Демон**. Эта карта всегда *врёт* и будет притворяться другой ролью, а также заставляет соседние слева и справа карты *врать*. Когда карта *врёт*, **Судья**, **Охотник** и **Шут** предоставляют неправильную информацию; **Рыцаря** можно будет убить; **Гренадер** не закончит игру вашим поражением.

Изначально карты появляются в закрытом виде, а у вас есть 9 жизней. Кроме того, в игре только один **Демон**.

За один ход игрок может выполнить одно из трёх действий:

1. Открыть неоткрытую карту. При этом игрок теряет 2 жизни.
2. Убить любую карту (открытую или закрытую):
 - Если игрок пытается убить **Демона**, то игрок побеждает.
 - Если игрок пытается убить **Рыцаря**, который врёт, то игрок теряет 5 жизней.
 - Если игрок пытается убить **Рыцаря**, который не врёт, то игрок не теряет жизней.
 - Если игрок пытается убить **Гренадера**, который не врёт, то игрок проигрывает.
 - Иначе игрок теряет 3 жизни.
3. Использовать способность открытой карты. За это действие жизни не тратятся.

Если после вашего хода у вас меньше одной жизни, вы проиграли. Иначе вы можете делать ходы дальше.



Формат входных данных

В первой строке каждой игры вводится единственное число n ($5 \leq n \leq 8$) — количество карт в игре.

В этой задаче четыре теста с разными n .

Тест	Ограничения
1	$n = 5$
2	$n = 6$
3	$n = 7$
4	$n = 8$

Протокол взаимодействия

Интерактор **не адаптивен**, то есть не пытается подстроить состояние игры против вас. Состояние игры предрешиено в начале игры и не меняется в зависимости от ваших действий.

Если в какой-то момент вы получили на вход слово «**stop**», это значит, что вы проиграли, и необходимо завершить программу. Например, у вас закончилось здоровье или вы сделали некорректный ход.

Вы можете использовать три команды:

1. **kill** x — вы убиваете карту под номером x (считая с единицы).
2. **unc** x — вы переворачиваете карту под номером x , если она не была перевёрнута, иначе ваш ход заканчивается. После этого действия вы получаете 2 урона.
3. **use** pos — вы используете способность **Судьи** или **Шута** на позиции pos, после этого вам предложат ввести одно или три числа для соответствующих ролей. Обратите внимание: если нужной роли на позиции pos не существует или она ещё не открыта, то игра сразу закончится.

Если была использована команда `kill`, вы получите сообщение «`win`», если вы убили Демона и нужно перейти к следующей игре, или «`miss x`» ($x < 0$), если вы этого не сделали: вы получаете $|x|$ урона.

Если была использована команда `unc`, вы получите информацию о выбранной карте:

- ‘K’ — Рыцарь;
- ‘H’ — Охотник;
- ‘S’ — Гренадер;
- ‘B’ — Шут;
- ‘J’ — Судья.

Обратите внимание, что ни одна карта не вскрыется сама как **Демон**.

Если вы получили на вход букву ‘H’, следом вы получите число `x` — расстояние, на котором находится Демон.

Если была использована команда `use`, вам нужно ввести одно или три числа для **Судьи** и **Шута** соответственно:

- Если была использована способность **Шута**, вернётся одно число — количество **Демонов** среди 3 выбранных карт, число от 0 до 3.
- Если была использована способность **Судьи**, вернётся одно число — 0, если карта врёт, 1, если карта не врёт.

Не забывайте сбрасывать буфер после каждого вывода. Для этого можете использовать `std::endl` в C++ или стандартную функцию `print` в Python. Не используйте `ios_base::sync_with_stdio(false)` в C++.

Пример

стандартный ввод	стандартный вывод
5	
miss -3	kill 2
H 2	unc 4
win	kill 1
...	...

Замечание

Что произошло в примере:

Первым действием мы убили карту на позиции 2, это был не **Демон** и не **Рыцарь**.

Вторым действием мы открыли карту на позиции 4, и это оказался **Охотник**, который сказал, что на расстоянии 2 находится **Демон**.

Третьим действием мы убили **Демона**; оказалось, что **Охотник** не соврал.