Предисловие:

Начните Ваш файл со скриптом с указания Вашей фамилии и инициалов:

var Name=”Фамилия И.О.”;

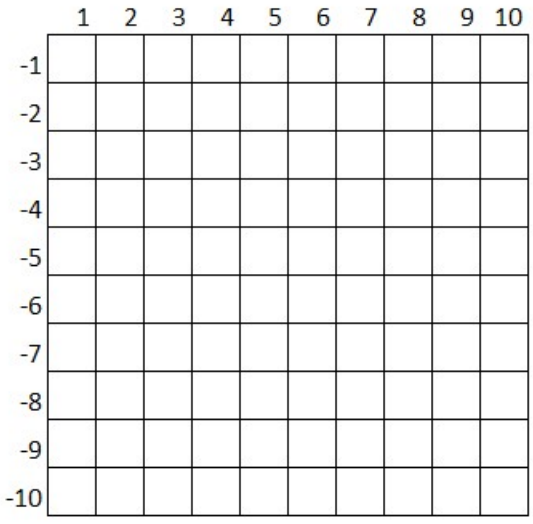
Все функции, которые Вам предстоит написать, принимают в себя 2 параметра: текущее состояние поля (state) и бортовой журнал (diary). Заголовки функций будут приведены в соответствующих разделах задач.

1 задача: функция для расстановки кораблей (*set*).

Функция получает состояние поля и бортовой журнал. Возвращает массив-координаты поля, в которых будет установлен корабль, а также бортовой журнал (с возможными изменениями).

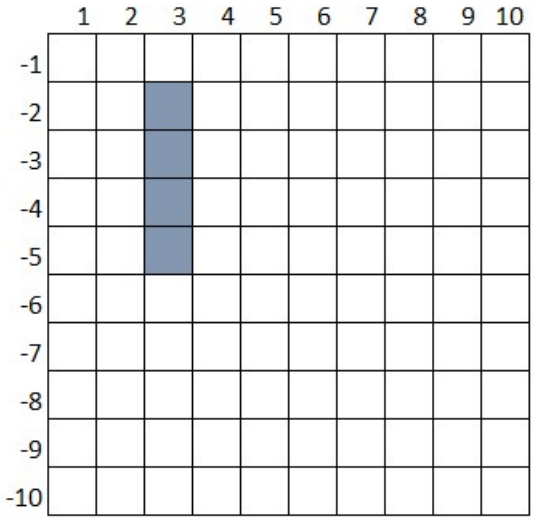
```
function set(state,diary){
    // Текст вашей функции
    return [y0,x0,y1,x1,diary];
}
```

Поле боя кодируется отрицательными значениями по вертикали и положительными – по горизонтали:



Поле задается в виде строки, состоящей из 100 символов – «нулей» и «единиц». Пустое поле – «0», заполненное палубой корабля –

«1». Допустим, я хочу поместить 4-палубный корабль строго вертикально со 2 строки по вертикали, с 3 столбца по горизонтали. Тогда описанная выше функция set(state,diary) должна вернуть массив [-2,3,-5,3]:



При этом состояние state, которое получит моя функция на следующем шаге, будет представлено строкой “00000000000010000000001000000000100000000000000000000000000000000000000000000000000000000000000000”, где 13, 23, 33 и 43 «нули» поменялись на «единицы» – кодирование поля идет поклеточно слева направо и сверху вниз.

Бортовой дневник – это Ваша персональная записная книжка длиной в 2048 символов, куда Вы можете записывать дополнительные данные о ходе боя или расстановки кораблей (внешняя память). Может понадобиться, если Вы планируете применять хитрые тактики (например, проверить, а не применял ли противник в прошлой партии те же самые приемы?). Самая обычная строка. Только избегайте двойных кавычек: при передаче в функцию они будут заменены на апострофы.

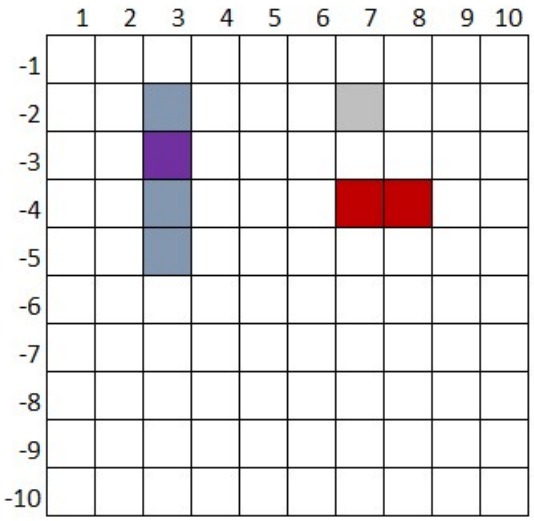
Ваша цель – сделать функцию универсальной, то есть чтобы с ее помощью можно было установить любое количество кораблей (желательно, все 10: 4 1-палубных, 3 2-палубных, 2 3-палубных, 1 4-палубный). Корабли не могут находиться в смежных ячейках по диагонали или прямой.

2 задача: функция для атаки вражеских кораблей (*hit*).

Функция получает состояние известных ячеек атакуемого поля и бортовой журнал. Возвращает массив из двух координат поля, в которые будет произведен выстрел, а также бортовой журнал (с возможными изменениями). Например, для клетки с координатами (-3,3) возвращаемое значение будет “[-3,3]”. Опять-таки, следует сделать функцию максимально универсальной, чтобы избежать штрафов (о них подробнее см. ниже).

```
function set(state,diary){
    // Текст вашей функции
    return [y0,x0,diary];
}
```

Рассмотрим следующую картину поля боя:



Цвет обозначает состояние и соответствующую цифру в строке:

- неизведанные клетки, по которым удар не наносился – белые, «0»;
- корабль ранен, как по адресу (-3,3) – сиреневый/темно-синий, «1»;
- корабль поражен, как по адресу (-4,7) или (-4,8) – бордовый, «2»;
- удар пришелся мимо, как по адресу (-2,7) – серый, «3».

Таким образом, поле выше можно закодировать строкой:
“0000000000000000300000100000000000002200000000000000000000000000000000000000000000000000000000000000000000000000000000”

Обратите внимание на код: самих «неоткрытых» клеток кораблей (например, клетки (-2,3), (-4,3) и (-5,3)) на такой карте, очевидно, не видно, как и положено в «Морском бою»: ведь Вы их атакуете. Они приведены на иллюстрации для удобства понимания.

Понятие штрафов:

За работу Вашего алгоритма начисляются очки. За каждый необдуманный шаг очки вычитаются. Ниже приведена сводная таблица штрафов:

Поражение	0
Неустановленный корабль 4	-1
Неустановленный корабль 3	-2
Неустановленный корабль 2	-3
Неустановленный корабль 1	-4
Неустановленный, неправильный корабль	-5
Выстрел по зоне, далекой от точек ранения	-6
Выстрел по зоне, где заведомо не могло быть корабля	-7
Выстрел за границу поля	-8
Повторный выстрел по "раненому" кораблю	-9
Повторный выстрел по "убитому" кораблю	-10
Выстрел по уже "простреленному мимо" полю	-11
Выстрел по одному и тому же полю дважды	-20
Победа	+21

«Выстрел по зоне, далекой от точек ранения» подразумевает атаку по удаленной от раненого корабля ячейке (*когда осуществляется продолжение «прострела» по случайным полям вместо «добивания» корабля*).

Примечание для владельцев демо-версии:

Если у Вас доступна демо-версия виртуального пространства (файл seashore.exe), то в каталоге «ships» должны быть размещены две копии Вашего файла с функциями set() и hit() : «fleet1.js» и «fleet2.js». Именно из этих двух файлов берет данные приложение “seashore.exe”. Игра автоматически проводится по адресу http://localhost:8080 после запуска приложения. Для сохранения результата прогона запустите приложение “seashore.exe” из консоли с перенаправлением в произвольный текстовый файл. Например:

D:\seashore> seashore.exe > test.txt

В файле test.txt, по завершении эмуляции, появятся данные о прошедшей игре.