

1.

### ШАХМАТЫ.

Поле - 8\*8 клеток, фигуры обозначаются буквами, (или каждая - двумя буквами, или белые - белая буква на черном фоне, черные - черная буква на белом фоне).

Рисуется поле. Человек играет с человеком.

Ходы пишутся в обычных обозначениях.

Программа должна отслеживать, чей ход и может ли фигура походить на выбранную клетку.

Рокировки и превращения пешек в ферзей отслеживать не обязательно.

В любой момент позицию можно записать в текстовый файл след. образом:

```
-- -- -- -- -- BR BK --
-- BP BP BV -- BP -- BP
-- -- -- BN -- -- BP --
-- -- -- WP BN WP -- WP
BR -- -- WP -- -- --
WP -- -- -- -- -- --
-- WB WP -- -- WP BQ --
-- -- WK WR -- -- -- WR
```

2.

### ШАШКИ.

Поле - 8\*8 клеток,

Шашки обозначаются буквами (или каждая - двумя буквами, или белые - белая '0' на черном фоне, черные - черная '0' на белом фоне).

Рисуется поле. Человек играет с человеком.

Ходы пишутся в обычных обозначениях.

Программа должна отслеживать, чей ход и может ли шашка походить на выбранную клетку.

Превращения шашек в дамки должно учитываться.

В любой момент позицию можно записать в текстовый файл след. образом:

```
- - - - - B - -
- - B - - - - -
- - - B - - - -
- - - - B - - -
- - - W - - - -
W - - - - - -
- W - - - - -
- - W - - - - -
```

#### 4. ХОД КОНЁМ (лабиринт)

Поле размера  $M \times N$ . Каждая позиция помечена или 0 (море) или буквой I (остров). В левом нижнем углу и в правом верхнем углу находятся острова. Задача - добраться с одного острова на другой, если следующим можно выбирать только остров, до которого можно добраться за один ход шахматного коня.

Размещение островов должно быть случайным и таким, чтобы хотя бы один путь существовал. Островов должно быть не слишком много и не слишком мало. (Чтобы было достаточно трудно найти путь.)

Перед началом игры должен быть выбор из двух вариантов игры: первый - количество ходов ограничено (например, количеством ходов в том варианте ответа, который использовался при составлении поля), второй - нельзя возвращаться на тот остров, с которого только что приплыл.

Во время игры, если нажать на F1, должны выделяться все острова, на которые можно попасть ходом коня. По клавише F10 игру можно в любой момент закончить, тогда последовательно выделяются острова, которые составляют одно из решений задачи.

#### 5. ЗЕМЛЕТРЯСЕНИЕ (лабиринт)

Поле размера  $M \times N$ . Препятствия отмечены буквой D. Лекарства - буквой o. Задача - добраться с левого нижнего угла до правого верхнего угла. После начала пути с неба начинают падать камни (новые препятствия). Изначально играющий, положение которого отмечается \*, имеет три жизни. Если камень упадет на него, на одну жизнь у него станет меньше. Если он съест лекарство, то число жизней увеличится. Размещение препятствий должно быть случайным и таким, чтобы хотя бы один путь существовал. Камней должно быть не слишком много и не слишком мало. (Чтобы было достаточно трудно найти путь, но была возможность остаться живым.)

Перед началом игры нужно иметь возможность задать уровень (чем выше уровень, тем чаще падают камни)

#### 3. КРЕСТИКИ-НОЛИКИ

Поле -  $N \times M$  клеток. Человек играет с компьютером. Кто начинает первым - задается в начале игры.

Программа должна сама регистрировать выигрыш. Должна быть реализована разумная стратегия игры компьютера.

#### 6. ПРОСТОЙ ЛАБИРИНТ

Поле размера  $M \times N$ .

Вертикальные стены лабиринта отмечены символом '|', горизонтальные - символом '-'.

Программа должна уметь хорошо генерировать такие лабиринты, чтобы, во-первых, существовал проход из входа к выходу и, во-вторых, стены ставились по возможности не хаотично, а образовывали тоннели, тупики и т.д.

При столкновении со стеной раздается звонок. После нажатия клавиши F10 ("сдаюсь") существующий путь выделяется.

7. САПЕР (Минное поле)

Поле - N\*M клеток.

Рисуется поле. На нем расставляются мины. Должно быть несколько уровней сложности.

Ход - постановка курсора на нужную клетку и нажатие клавиши, означающий, что нужно сделать, поставить мину или раскрыть клетку. Раскрытие уже раскрытой клетки означает «раскрыть все оставшиеся клетки вокруг нее».

8. LINES.

Поле - 9\*9 клеток.

Рисуется поле. В начале - 5 шариков. После каждого хода - по 3.

Должна быть проверка, есть ли туда путь. Если образовалось 5 шариков одного цвета в линию (включая расположение по диагонали), то шарики должны исчезнуть. За каждую исчезнувшую линию даются очки.

9. Quattropoli

Поле - 7 колонок, 6 линий

Рисуется поле. Представляем, что это поле расположено вертикально и два игрока (один из них - компьютер) по очереди кидают сверху жетон своего цвета. Он останавливается на первой свободной позиции в этой колонке. Выигрывает тот, у кого первого будет 5 подряд (вертикально, горизонтально или по диагонали) жетонов его цвета.

Ход - установка курсора под колонку и нажатие спец.клавиши.

Должна быть проверка, когда кто-то выиграл.

Также, если возможна на данном ходу победа, то компьютер должен это отслеживать.

10. Blocks

Поле NxM (N и/или M четное). На нем расположены парные картинки (изображения не видно). Нужно попарно открывать картинки. Если они одинаковые, то они остаются открытыми. Если разные, то они обратно переворачиваются.

Ход - задание двух картинок. Окончание игры - когда все картинки открыты.

Нужно спрашивать у человек его имя, записывать его результат (число ходов), выдавать список игравших на экран, отмечать там текущего игрока и выдавать фразу, соответствующую успешности игры (хвалить, ругать, подбадривать).