ОСНОВНЫЕ ПРАВИЛА ИГРЫ:

**Основные требования**

- Игра ведется пошагово;

- В игре имеется лабиринт (прямоугольная область, состоящая из клеток), в котором расставлены следующие игровые элементы: умный робот, глупый робот, вертикальные и горизонтальные стены (располагается на стыке клеток), *ландшафты (*обычные болота*, расширяемые болота, пески)*, точка выхода;

- Игра считается пройденной, если умный робот находится в клетке с точкой выхода. Игра считается проигранной, если умный робот находится в клетке с глупым роботом;

*Требования к расширению игры:*

*- В игре есть ландшафты, которые влияют на перемещение обоих роботов (количество ходов или пропусков и направления движения).*

*- В игре есть четыре сезона года (Зима, Весна, Лето, Осень), которые сменяются каждые 3 хода. Сезоны года влияют на состояние ландшафтов, размещённых в игровом поле.*

**Требования к умному роботу**

- Игрок управляет умным роботом, перемещая его в соседние клетки (вправо, влево, вверх, вниз);

- Для умного робота препятствиями являются стены, болота;

- Если умный робот попал в пустую клетку, на которую расширится болото, то умный робот погибает.

**Требования к глупому роботу**

- Компьютер управляет глупым роботом, перемещая его в соседние клетки (вправо, влево, вверх, вниз);

- Глупый робот имеет две модели поведения в зависимости от нахождения умного робота внутри своей зоны видимости (прямоугольная область 5х5 с центром в позиции глупого робота);

- Глупый робот идёт по эффективному маршруту до умного робота, если умный робот находится в зоне видимости глупого робота;

- Глупый робот перемещается вправо или влево по направлению к умному роботу, если умный робот находится вне зоны видимости глупого робота;

- Для глупого робота препятствиями являются стены;

- Глупый робот может заходить в болото, но он пропускает 3 хода;

- Если глупый робот попал в болото, которое после периода осени должно уйти, то глупый робот перестаёт пропускать ходы после исчезновения болота.

***Требования к ландшафтам:***

- Любое болото замерзает в период зимы. По замерзшему болоту роботы передвигаются на две клетки до препятствия;

- Расширяемое болото в период осени расширяется вертикально и горизонтально на одну клетку. После периода осени расширяемое болото возвращается к исходным размерам;

- Любой робот при попадании в песок пропускает один ход.

СЦЕНАРИИ:

1. Основной сценарий «Играть».
2. **По указанию пользователя**, *игра* стартует.
3. **По указанию** *игры*, *поле* создаёт *ячейки* и формирует из них себя. При этом *поле* узнает позицию для *точки выхода* у *проектировщика лабиринтов*.
4. **По указанию** *игры*, *проектировщик лабиринтов* создаёт и размещает на *поле* *игровые элементы*: *стены*, *ландшафты*, *умного робота*, *глупого робота*.
5. **Делать**
   1. **По указанию пользователя**, *умный робот* перемещается на соседнюю *ячейку*. [(дочерний сценарий)](#сцен_2)
   2. **По указанию** *игры*, *глупый робот* проверяет наличие *умного робота* в своей *ячейке*. [(дочерний сценарий)](#сцен_3)
   3. **По указанию** *игры*, *глупый робот* перемещается на соседнюю *ячейку*. [(дочерний сценарий)](#сцен_4)
   4. **По указанию** *игры*, *глупый робот* проверяет наличие *умного робота* в своей *ячейке*. [(дочерний сценарий)](#сцен_3)
   5. **По указанию** *игры*, текущий *сезон года* сменяется на следующий, если текущий *сезон года* продлился 3 хода.
   6. По указанию *игры*, *поле* обновляет свои *ландшафты* [(дочерний сценарий)](#сцен_6).
   7. **По указанию** *игры*, *умный робот* проверяет наличие *болота* в своей *ячейке*. [(дочерний сценарий)](#сцен_7)

**Пока** *игра* видит *умного робота* на *поле*.

1. *Игра* считает победителем *умного робота*, если он находится внутри *точки выхода*. Иначе *игра* считает проигравшим *умного робота*, если он находится внутри *глупого робота*.
2. **Сценарий завершается.**
3. Дочерний сценарий «*умный робот* перемещается на соседнюю пустую *ячейку*». [(родительский сценарий)](#сцен_1__4_1)
4. **В ответ на запрос** *умного робота*, собственная *ячейка* **сообщает**, что *стены* нет в направлении движения *умного робота*.

[(альтернативный сценарий – умный робот утыкается в стену)](#сцен_2_1)

1. **В ответ на запрос** *умного робота*, собственная *ячейка* **сообщает** о соседней *ячейке* в направлении движения *умного робота*.

[(альтернативный сценарий – умный робот утыкается в край поля)](#сцен_2_2)

1. **В ответ на запрос** *умного робота*, соседняя *ячейка* **сообщает**, что в ней отсутствует *болото*.

[(альтернативный сценарий – умный робот утыкается в болото)](#сцен_2_3)

1. **По указанию** *умного робота*, собственная *ячейка* извлекает *умного робота* из себя.
2. **По указанию** *умного робота*, соседняя *ячейка* помещает его в себя.

[(альтернативный сценарий – умный робот попадает в точку выхода)](#сцен_2_4)

1. **Сценарий завершается.**
   1. Альтернативный сценарий «*умный робот* утыкается в *стену*».

Сценарий выполняется с пункта 1 сценария 2

[(место ответвления)](#сцен_2_пункт_1) [(исходный сценарий)](#сцен_2)

1. **В ответ на запрос** *умного робота*, собственная *ячейка* **сообщает**, что *стена* имеется в направлении движения *умного робота*.
2. *Умный робот* решает, что он не может переместиться в соседнюю *ячейку*.
3. **Сценарий завершается.**
   1. Альтернативный сценарий «*умный робот* утыкается в край *поля*».

Сценарий выполняется с пункта 2 сценария 2

[(место ответвления)](#сцен_2_пункт_2) [(исходный сценарий)](#сцен_2)

1. **В ответ на запрос** *умного робота*, собственная *ячейка* **сообщает**, что по данному направлению движения *умного робота* нет соседней *ячейки*.
2. *Умный робот* решает, что он не может переместиться в соседнюю *ячейку*.
3. **Сценарий завершается.**
   1. Альтернативный сценарий «*умный робот* утыкается в элемент ландшафта».

Сценарий выполняется с пункта 3 сценария 2

[(место ответвления)](#сцен_2_пункт_3) [(исходный сценарий)](#сцен_2)

1. **В ответ на запрос** *умного робота*, соседняя *ячейка* **сообщает**, что в ней находится *болото*.
2. *Умный робот* решает, что он не может переместиться в соседнюю *ячейку*.
3. **Сценарий завершается.**
   1. Альтернативный сценарий «*умный робот* попадает в *точку выхода*».

Сценарий выполняется c пункта 5 сценария 2

[(место ответвления)](#сцен_2_пункт_5) [(исходный сценарий)](#сцен_2)

1. **По указанию** *умного робота*, соседняя *ячейка* помещает его в себя.
2. *Точка выхода* убирает *умного робота* с *поля*.
3. *Точка выхода* сообщает о нахождении *умного робота* *игре*.
4. **Сценарий завершается.**
5. Дочерний сценарий «*глупый робот* проверяет наличие *умного робота* в своей *ячейке*». [(родительский сценарий)](#сцен_1__4_2)
6. **В ответ на запрос** *глупого робота*, собственная *ячейка* **сообщает**, что она не содержит *умного робота* в себе.

[(альтернативный сценарий – умный робот находится в собственной ячейке глупого робота)](#сцен_3_1)

1. **Сценарий завершается.**
   1. Альтернативный сценарий «*умный робот* находится в собственной *ячейке* *глупого робота*».

Сценарий выполняется с пункта 1 сценария 3

[(место ответвления)](#сцен_3_пункт_1) [(исходный сценарий)](#сцен_3)

1. **В ответ на запрос** *глупого робота*, собственная *ячейка* **сообщает**, что она содержит *умного робота* в себе.
2. *Глупый робот* убирает *умного робота* с *поля*.
3. *Глупый робот* сообщает о нахождении *умного робота* *игре*.
4. **Сценарий завершается.**
5. Дочерний сценарий «*глупый робот* перемещается на соседнюю *ячейку*». [(родительский сценарий)](#сцен_1__4_3)
6. *Глупый робот* проверяет, что он не пропускает ход.

[(альтернативный сценарий – глупый робот пропускает ход)](#сцен_4_1)

1. **В ответ на запрос** *глупого робота*, *поле* **сообщает** о всех *ячейках* из заданной прямоугольной области видимости *глупого робота*.
2. *Глупый робот* определяет, что *умный робот*, находится вне своей области видимости по его наличию в полученных *ячейках*.

[(альтернативный сценарий – глупый робот внутри своей области видимости умного робота)](#сцен_4_2)

1. **В ответ на запрос** *глупого робота*, *поле* **сообщает** о позиции *ячейки* *умного робота*.
2. *Глупый робот* решает, двигаться ли ему вправо или влево по направлению движения к *умному роботу*.

[(альтернативный сценарий – умный робот находится в одном столбце с глупым роботом и вне области его видимости)](#сцен_4_3)

1. **В ответ на запрос** *глупого робота*, собственная *ячейка* **сообщает** об отсутствии *стены* по направлению движения.

[(альтернативный сценарий – глупый робот утыкается в стену)](#сцен_4_4)

1. **В ответ на запрос** *глупого робота*, собственная *ячейка* **сообщает** о соседней *ячейке*.
2. **По указанию** *глупого робота*, собственная *ячейка* извлекает *глупого робота* из себя.
3. **По указанию** *глупого робота*, соседняя *ячейка* помещает его в себя.
4. **В ответ на запрос** *глупого робота*, новая собственная *ячейка* **сообщает**, что на данной *ячейке* нет некоторого *ландшафта*.

[(альтернативный сценарий – глупый робот попадает в ландшафт)](#сцен_4_5)

1. **Сценарий завершается**.
   1. Альтернативный сценарий «*глупый робот* пропускает ход».

Сценарий выполняется с пункта 1 сценария 4

[(место ответвления)](#сцен_4_пункт_1) [(исходный сценарий)](#сцен_4)

1. *Глупый робот* проверяет, что он пропускает данный ход.
2. *Глупый робот* убавляет количество пропускаемых ходов.
3. *Глупый робот* решает, что он не может двигаться.
4. **Сценарий завершается.**
   1. Альтернативный сценарий «*глупый робот* видит внутри своей области видимости *умного робота*».

Сценарий выполняется с пункта 3 сценария 4

[(место ветвления)](#сцен_4_пункт_3) [(исходный сценарий)](#сцен_4)

1. *Глупый робот* определяет, что *умный робот*, находится в своей области видимости по его наличию в полученных *ячейках*.
2. **В ответ на запрос** *глупого робота*, *поле* **сообщает** кратчайший маршрут от *ячейки* *глупого робота* до *ячейки* *умного робота*.  
   (альтернативный сценарий – нет маршрута от ячейки глупого робота до умного робота)
3. **По указанию** *глупого робота*, собственная *ячейка* извлекает *глупого робота* из себя.
4. **По указанию** *глупого робота*, соседняя *ячейка* помещает его в себя.
5. **В ответ на запрос** *глупого робота*, новая собственная *ячейка* **сообщает**, что на данной *ячейке* нет некоторого *ландшафта*.

[(альтернативный сценарий – глупый робот попадает в ландшафт)](#сцен_4_5)

1. **Сценарий завершается.**
   1. Альтернативный сценарий «*умный робот* находится в одном столбце с *глупым роботом* и вне области его видимости».

Сценарий выполняется с пункта 5 сценария 4

[(место ветвления)](#сцен_4_пункт_5) [(исходный сценарий)](#сцен_4)

1. *Глупый робот* видит, что он расположен на одном столбце ячеек с *умным роботом*.
2. *Глупый робот* решает, что он не может переместиться в соседнюю *ячейку*.
3. **Сценарий завершается.**
   1. Альтернативный сценарий «*глупый робот* утыкается в *стену*».

Сценарий выполняется с пункта 6 сценария 4

[(место ветвления)](#сцен_4_пункт_6) [(исходный сценарий)](#сцен_4)

1. **В ответ на запрос** *глупого робота*, собственная *ячейка* **сообщает** об отсутствии *стены* по направлению движения.
2. *Глупый робот* решает, что он не может переместиться в соседнюю *ячейку*.
3. **Сценарий завершается.**
   1. Альтернативный сценарий «некоторый робот попадает в некоторый ландшафт».

Сценарий выполняется с пункта 9 сценария 4

[(место ветвления)](#сцен_4_пункт_10) [(исходный сценарий)](#сцен_4)

Сценарий выполняется с пункта 7 сценария 4.2

[(место ветвления)](#сцен_4_5_пункт_7) [(исходный сценарий)](#сцен_4_2)

1. **В ответ на запрос** некоторого *робота*, новая собственная *ячейка* **сообщает**, что на данной *ячейке* есть *болото*.
2. *Глупый робот* решает пропустить 3 хода.
3. **Сценарий завершается.**
4. Альтернативный сценарий «Досрочное завершение игры». Сценарий выполняется в любой точке основного сценария.
5. **По указанию пользователя**, программа завершается без определения победителя.
6. **Сценарий завершается.**
7. Дочерний сценарий «поле обновляет свои ландшафты». [(родительский сценарий)](#сцен_1__4_6)
8. **По указанию** *поля*, *ландшафты* обновляют своё состояние в зависимости от нового *сезона года*.
9. **Сценарий завершается.**
10. Дочерний сценарий «умный робот проверяет наличие болота в своей ячейке». [(родительский сценарий)](#сцен_1__4_6)
11. **В ответ на запрос** *умного робота*, собственная *ячейка* **сообщает** об отсутствии *болота* внутри себя.

(альтернативный сценарий — умный робот находится в ячейке с болотом)

1. **Сценарий завершается.**

7.1) Альтернативный сценарий «некоторый робот попадает в некоторый ландшафт».

Сценарий выполняется с пункта 9 сценария 4

[(место ветвления)](#сцен_4_пункт_10) [(исходный сценарий)](#сцен_4)

1. **В ответ на запрос** некоторого *робота*, новая собственная *ячейка* **сообщает**, что на данной *ячейке* есть *болото*.
2. *Глупый робот* решает пропустить 3 хода.
3. **Сценарий завершается.**

СЛОВАРЬ ПРЕДМЕТНОЙ ОБЛАСТИ:

1. Игра.

Роль:

- инициирует создание основных игровых сущностей;

- определяет окончание игры и конечного победителя.

Знает:

- про игровое поле;

- какой проектировщик лабиринтов будет использоваться для заполнения поля.

1. Поле.

Роль:

- прямоугольная область, состоящая из ячеек.

Знает:

- про свои ячейки;

- про точку выхода;

Умеет:

- возвращать ячейки из заданного прямоугольного сектора;

- возвращать умного робота;

- возвращать глупого робота;

- возвращать точку выхода.

1. Проектировщик лабиринтов.

Роль:

- занимается заполнением игрового поля.

Знает:

- про позицию ячейки точки выхода;

- про позиции всех расставляемых игровых элементов.

Умеет:

- помещать в ячейки поля игровые элементы;

- помещать стены между ячейками.

1. Ячейка.

Роль:

- определяет позицию игрового элемента на поле;

- предоставлять информацию о своих соседях по горизонтали и вертикали.

Знает:

- про игровые элементы, которые в ней хранятся;

- про соседние ячейки;

- про соседние стены.

Умеет:

- возвращать ячейку-соседа по заданному направлению;

- возвращать все соседние ячейки, в направлении которых нет стен.

1. Точка выхода наследуется от Ячейки.

Роль:

- является разновидностью ячейки;

- является местом, куда нужно попасть умному роботу.

Знает:

- про своих слушателей, которым отправляет сообщение о «телепортировании» умного робота.

1. Игровой элемент:

Роль:

- является обобщением понятий: умный робот, глупый робот, болото.

Знает:

- о своей ячейке;

1. Умный робот наследуется от Игрового элемента.

Роль:

- игровой персонаж, которым управляет пользователь.

Знает:

- о своих препятствиях: стена, болото.

Умеет:

- пошагово перемещаться по полю по заданному пользователем направлению.

1. Глупый робот наследуется от Игрового элемента.

Роль:

- игровой персонаж, который пытается поймать умного робота.

Знает:

- о поле, в котором глупый робот находится;

- о своём текущем количестве пропускаемых ходов;

- о своей области видимости;

- про умного робота, который находится внутри него;

- о своих препятствиях: стена, болото;

- о пропуске 3-х ходов при попадании в болото.

Умеет:

- пошагово перемещаться по полю по заданному им же самим направлению;

- определять направление движения в зависимости от наличия умного робота в своей области видимости;

- определять наличие умного робота в своей области видимости.

1. Сезон года.

Роль:

- определяет текущее состояние ландшафтов на поле.

1. Ландшафт.

Роль:

- является препятствием.

Знает:

- о своих элементах ландшафта.

Умеет:

- менять своё состояние в зависимости от текущего сезона года;

- говорить обоим роботам как нужно себя вести при попадании в себя (пропуск ходов или совершение ходов заданного количества и по заданному направлению).

1. Элемент ландшафта наследуется от Игрового элемента.

Роль:

1. Болото наследуется от Ландшафта.

Роль:

- препятствие для умного робота. Если же умный робот попадает в болото (например, расширяемое болото расширилось в ячейку с умным роботом), то умный робот погибает;

- место пропуска 3-х ходов для глупого робота.

1. Расширяемое болото наследуются от Ландшафта.

Роль:

- расширятся вертикально и горизонтально на одну клетку.

Знает:

- о болотах размещаемых на своей ячейке и её соседях.

1. Песок наследуются от Ландшафта.

Роль:

- место пропуска 1-го хода для обоих роботов.