# Формулировка задания

Игра "Коза и капуста".

Правила игры:

* имеется загон NxM клеток, в котором находится коза, капуста, ящики и стены (располагаются внутри клеток);
* ящики и стены, возможно, образуют непроходимый лабиринт;
* цель козы — добраться до капусты, двигая ящики;
* коза может двигать только один ящик от себя или на себя;
* количество шагов козы ограничено.

Дополнительные требования:

* необходимо предусмотреть в программе точки расширения, используя которые можно реализовать вариативную часть программы (в дополнение к базовой функциональности).

Вариативность:

* возможны ящики с различным видом взаимодействия (зацепления) между собой. Если несколько ящиков сцепляются, то коза толкает и отталкивает сразу несколько ящиков (если это возможно). Ящики разного вида должны быть визуально различимы.

Реализовать:

* магнитные ящики, которые притягиваются или отталкиваются в зависимости от полюса (стороны ящика). Область действия магнитного поля - соседняя клетка. Отталкивающиеся ящики не могут находиться в соседних клетках;
* металлический ящик, который притягивается к магнитному ящику.

### 

# Формулировка упрощённого варианта задания

Игра "Коза и капуста".

Правила игры:

* имеется загон NxM клеток, в котором находится коза, капуста, ящики и стены (располагаются внутри клеток);
* ящики и стены, возможно, образуют непроходимый лабиринт;
* цель козы — добраться до капусты, двигая ящики;
* коза может двигать только один ящик от себя или на себя;
* количество шагов козы не ограничено.

### Сценарии

**1) Сценарий** «Играть»

1. **По указанию пользователя,** Игра стартует.
2. **По указанию** Игры, Поле создаёт Ячейки и формирует из них себя.
3. **По указанию** Игры, Лабиринт создаёт и размещает на Поле Стены, Ящики, Козу и Капусту.
4. **В ответ на запрос** Игры, Поле **сообщает** о Козе, которая находится на нём.
5. **Делать**
   1. **По указанию** **пользователя,** Коза перемещается на соседнюю Ячейку с заёмом своей энергии.
   2. **Если** Коза перемещается на Капусту, то Коза съедает Капусту, что приводит к исчезновению Капусты с Поля.
   3. **В ответ на запрос** Игры, Поле **сообщает** о Козе, которая находятся на нём.

**Пока** Коза имеет энергию и не съела капусту.

1. Игра считается выигранной, если Коза съела Капусту.
2. Игра считается проигранной, если у Козы была израсходована вся энергия, при этом Капуста не была ею съедена.
3. **Сценарий завершается**.

**2) Дочерний сценарий «**Лабиринт создаёт и размещает на Поле Стены, Ящики, Козу и Капусту**»**

1. Лабиринт создаёт и расставляет последовательности Стен внутри Поля.
2. Лабиринт создаёт и расставляет последовательности Ящиков внутри Поля.
3. Лабиринт создаёт Козу и наделяет её энергией.
4. Лабиринт помещает на Поле Козу.
5. Лабиринт создаёт Капусту и помещает её на Поле.
6. **Сценарий завершается**.

**3) Дочерний сценарий** «Коза перемещается на соседнюю Ячейку с заемом своей энергии»

1. **В ответ на запрос** Козы, текущая Ячейка сообщает о соседней по направлению движения Козы Ячейке.
2. **В ответ на запрос** Козы, соседняя Ячейка сообщает о возможном Ящике на ней.
3. **В ответ на запрос** Козы, Ящик по возможности двигается в направлении желаемого движения Козы.
4. **В ответ на запрос** Козы, соседняя Ячейка сообщает, что является свободной.
5. Коза решает, что она может переместиться в соседнюю Ячейку.
6. **По указанию** Козы, её собственная энергия расходуется на величину, необходимую для перемещения, т.к. у неё энергии достаточно.
7. **По указанию** Козы, Ячейка извлекает Козу из себя.
8. **По указанию** Козы, соседняя Ячейка помещает её в себя, т.к. она свободна.
9. **Сценарий завершается**.

**3.1) Альтернативный сценарий** «Соседняя по направлению движения Козы Ячейка не является свободной». (продолжение с п. 3)

1. **В ответ на запрос** Козы, соседняя Ячейка **сообщает**, что не является свободной.
2. Коза решает, что она не может переместиться в соседнюю Ячейку.
3. **Сценарий завершается**.

**4) Дочерний сценарий** «Коза съедает Капусту, что приводит к исчезновению Капусты с Поля».

1. **В ответ на запрос** Козы, Ячейка **удаляет** Капусту из себя.
2. Коза становится сытой.
3. **Сценарий завершается**.

**5) Дочерний сценарий** «Игра считается выигранной, если Коза достигла ячейку с Капустой»

1. Игра **запрашивает** у Козы, сыта ли она.
2. Коза **сообщает** Игре, что она сыта.
3. Игра считается выигранной.
4. **Сценарий завершается.**

**6) Дочерний сценарий** «Игра считается проигранной, если у Козы была израсходована вся энергия, при этом Капуста не была ею достигнута».

1. Игра **запрашивает** у Козы, сыта ли она.
2. Коза **сообщает** Игре, что она не сыта.
3. Игра считается проигранной.
4. **Сценарий завершается.**

**7) Альтернативный сценарий** «Досрочное завершение игры». Сценарий **выполняется в любой точке** главного сценария

1. **По указанию пользователя,** программа завершается без определения исхода игры.
2. **Сценарий завершается.**

### Словарь предметной области

**Игра** - управляет игровым циклом: инициирует создание всех сущностей, определяет окончание и исход игры.

**Поле** - прямоугольная область, состоящая из Ячеек.

**Ячейка** - квадратная область Поля. Знает о четырёх соседних Ячейках. На ней может располагаться одна Стена, или один Ящик, или одна Коза, или одна Капуста. Говорит, свободно ли в заданном направлении.

**Стена** - непроходимое Препятствие для Козы и Ящиков, располагающееся на Ячейке.

**Лабиринт** - умеет создавать Стены, Ящики, Козу и Капусту и размещать их на Поле в Ячейках. Позиции этих сущностей Лабиринт определяет самостоятельно.

**Коза** - умеет однократно перемещаться в течение своего хода. Коза перемещается в соседнюю Ячейку, затрачивая собственную энергию, но не может пройти через Стену или неподвижный Ящик. Но Коза может толкать либо тянуть один Ящик. В течение своего хода Коза может съесть Капусту из Ячейки, в которой находится. Коза знает о том, съела ли она Капусту.

**Ящик** – может быть сдвинут Козой на одно из 4 направлений в случае, если соседняя Ячейка свободна.

**Капуста** - съедается Козой, находящейся на ней.

**Направление** - может быть Север, Юг, Запад или Восток.