**Описание алгоритма.**

Алгоритм основан на случайном выборе карт с увеличением вероятности выпадения предпочтительных карт в зависимости от карт, которые не использовал противник. Причём стратегия выбора карт для атаки и защиты различаются.

То есть мы используем функцию ИИ = (карты игрока, карты ИИ, тип хода – атака или защита) => { return выбранная карта}

Карты малого достоинства – 0,1,2,3. Среднего – 4,5,6,7,8. Большого – 9, 10, 11.

Логика выбора предпочтительных карт:

1. Атака.
   * Атаковать большими картами больше смысла, если у противника мало карт большого и среднего достоинства – т.е. так шанс нанести больше урона увеличивается.
   * Имеет смысл атаковать картами малого достоинства, особенно в начале игры, поскольку противник не всегда ожидает такого хода и может потратить на защиту от маленькой карты большую или среднюю. Особенно выгодно атаковать с 0.
2. Защита.
   * Защищаться с 0 не имеет никакого смысла, поскольку шанс того, что противник атакует с 0 мал. Напротив – атакуя с нуля, шанс того, что враг потратит на защиту карту больше нуля высок.
   * Защищаться с карт высокого достоинства тоже маловыгодно, но в тоже время стоит учитывать наличие у противника таких карт.

Исходя из вышеизложенного, для атаки в начале игры следует с большей вероятностью выбирать карты малого достоинства и среднего. Для защиты – средние, причём следует исключить использование 0 и 1 в защите.

**Для реализации этой идеи мы можем каждой карте AI присвоить некоторое число, которое будет отражать шанс её выпадения относительно других карт, назовем эти числа числами относительной вероятности.** Например, для первого хода атаки – 0 – шанс 4, 1..5 – шанс 3, 6..9 – шанс 2, 10 и 11 шанс 1.

**Как используются числа относительной вероятности?**

1. Для данного хода для всех карт ИИ формируем массив объектов вида {число карты, относительная вероятность этого числа}.
2. Передаём этот массив в функцию, function randCard(cards), которая определит карту для хода ИИ.

**Как работает randCards?**

1. Создаём массив A, в который каждая карта записывается n раз, где n = ранее определённому числу относительной вероятности.
2. Вычисляем случайное целое число Ind в диапазоне индексов созданного массива.
3. Возвращаем A[Ind].

**Как меняются числа относительной вероятности?**

1. Вычисляются две переменные для карт среднего и высокого достоинства. Значения переменных - количество отсутствующих карт соответствующего достоинства.
2. Эти числа, делённые на некоторый коэффициент прибавляются к начальному значению вероятностного числа. Например,  if (i>=6 && i <= 9) cards.push(new InitCard(i,1 + Math.ceil((deciderHigh+deciderMidlle)/2)))