

Тасуем колоду карт

Стандартная игральная колода состоит из 52 карт. Каждая карта соответствует одной из четырех мастей (пики, червы, бубны и трефы) и одному из 13 номиналов (от 2 до 10, валет (J), дама (Q), король (K) и туз (A)).

Таким образом, каждая игральная карта может быть представлена при помощи двух символов. Первый из них указывает на номинал карты (от 2 до 9, T (десятка), J, Q, K или A), а второй – на масть (s = пики (spades), h = червы (hearts), d = бубны (diamonds) и c = трефы (clubs)). В табл. представлены некоторые из возможных обозначений игровых карт.

Карта	Обозначение
Валет пик	Js
Двойка треф	2c
Десятка бубен	Td
Туз червей	Ah
Девятка пик	9s

Начните с написания функции `createDeck`. В ней должны использоваться циклы для создания полной колоды карт путем сохранения в список двухсимвольных аббревиатур всех 52 карт. Именно этот список и будет возвращаемым из данной функции значением. На вход функция `createDeck` принимать параметры не будет.

Напишите вторую функцию с именем `shuffle`, которая будет случайным образом перетасовывать карты в списке. Одна из техник тасования колоды заключается в проходе по элементам и перестановке их с любым другим случайным элементом в этом списке. Вы должны создать свой собственный цикл для тасования карт в колоде, а не пользоваться стандартной функцией `shuffle` языка Python.

Используйте обе созданные функции в основной программе, в которой должна отображаться колода карт до и после тасования. Убедитесь, что основная программа выполняется только в случае, если файл не импортирован в качестве модуля.

Примечание. Хороший алгоритм тасования игральной колоды должен быть беспристрастным, что означает равную вероятность расположения каждой из карт в колоде после тасования. Однако алгоритм, предложенный в этом упражнении и предполагающий обмен позициями между каждой из карт в колоде с любой другой случайной. Списки 105 картой, не является таковым. В частности, карты, которые появляются позже в исходном списке, с большой вероятностью окажутся ближе к концу и в перетасованном списке. Как это ни странно, беспристрастной будет версия алгоритма, в которой при последовательном проходе по элементам каждый из них будет меняться позициями не со случайным элементом из всего списка, а со случайным элементом в диапазоне от позиции текущей карты и до конца колоды

Раздача карманных карт

Во многих карточных играх после процедуры тасования колоды каждый игрок получает на руки определенное количество карт. Напишите функцию `deal`, принимающую на вход три параметра: количество игроков, количество раздаваемых каждому из них карт и саму колоду. Функция должна возвращать список рук, которые были розданы игрокам. При этом каждая рука, в свою очередь, тоже является списком из входящих в нее карт. Во время раздачи карт игрокам функция параллельно должна удалять розданные карты из переданной ей третьим параметром колоды. Также принято раздавать карты каждому игроку по одной строго по очереди. Придерживайтесь этих принципов и при написании своей функции. Воспользуйтесь своими наработками из прошлого упражнения при построении структуры основной программы. Вам необходимо создать колоду карт, перетасовать ее и раздать четырем игрокам по пять карт. Выведите на экран карманные карты всех игроков, находящихся в раздаче, а также оставшиеся в колоде карты