ОПИСАНИЕ РЕАЛИЗАЦИИ

Сначала идёт импорт библиотек, опередиление констант и запуск фоновой музыки.

Затем функции: *terminate(), load\_background(name, colorkey=None), load\_image(name, colorkey=None), start\_screen(), finish\_screen()*, *card\_choice()*. Рассмотрим работу каждой из них.

Функция *terminate()* отвечает за выход из программы при нажатии красного крестика.

Функции *load\_background(name, colorkey=None)* и *load\_image(name, colorkey=None)* имеют одинаковый принцип работы и различаются только тем, что первая загружает стиль карт из папки data, а вторая – фоны из папки pics. Это было сделано для моего удобства – меня путало то, что карты и фоны лежат в одной папке.

После идёт функция *start\_screen(),* которая вызывается из главной функции *main()* и отвечает за красивую картинку на начальном экране.

Затем – функция *finish\_screen()*, которая также вызывается из главной функции *main(),* когда пасьянс разложен. Отвечает за надпись *«Congratulations, You Won!»*, время, за которое был разложен пасьянс и возможность начать новую игру, не закрывая окна.

Следующая – функция *card\_choice()*, которая должна позволять пользователю самому выбрать, сколько карт будет открываться за раздачу. Но что-то пошло не так, а время уже поджимало, поэтому выбор количества карт происходит рандомным образом, в функции *main()*.

Далее идут классы.

Класс *MovedCard(object)* отображает передвижение карт на экране. В нём инициализируются флаг *self.moved,* равный *True*, когда, хоть какая-то карта на экране, движется. Его функция *click\_up(self, deck\_list)* запускается после того, как пользователь отпускает левую кнопку мыши и проверяет, возможно ли вообще положить карту на другую. Функция *draw(self, screen, card\_dict)* отображает перемещение карты на экране.

Класс *Deck* – это родительский класс, проверяющий, попал ли игрок по карте или нет.

Класс *Deck1* – сыновний от класса *Deck.* Он позволяет класть карты в ячейках друг на друга только в правильной последовательности, чередуя цвета, а именно: король, дама, валет, десять, девять, восемь, семь, шесть, пять, четыре, три, два, туз.

Класс *Deck2* – это рисовка колоды. Переменная *kol*, которую мы опредилили ранее, применяется именно здесь – при нажатии на колоду, если она равна одна, открывается одна карта, а если трём – то три.

Класс *Deck3* – это «дома». Он отвечает за возможность положить друг на друга карты только одной масти в порядке: туз, два, три, четыре, пять, шесть, семь, восемь, девять, десять, валет, дама, король.

После классов снова идут функции.

*Shuffle\_cards()* отвечает за перемешивание карт и их распредиление на экране рандомным образом.

И, наконец, *main()* отвечает за основной цикл и нормальную работу всей программы. В ней:

* Опредиляются и рассчитываются переменные *vremya* и *kol*, отвечающие, соответственно, за время, за которое пользователь раскладывает пасьянс, и количество карт в раздаче;
* Загружаются карты;
* Рисуются и реализуются кнопки, отвечающие за быстрый рестарт игры и отключение звука фоновой музыки;
* Опредиляются 12 рабочих зон – колода, 7 обычных ячеек и 4 «дома», в которых происходит работа с картами

*DECK3*

*DECK2*

*DECK1*

*DECK*

*MOVEDCARD*