

Министерство науки и высшего образования РФ  
федеральное государственное автономное образовательное учреждение  
высшего образования  
«Омский государственный технический университет»

Лицей Академии Яндекса

Разработка и реализация игры 2048

Выполнили:  
Рогоза Владислав Юрьевич

Омск, 2023

## РЕФЕРАТ

Отчет 10 с., 3 рис., 0 табл., 3 источн., 1 прил.

ПРОЕКТ, РАЗРАБОТКА ИГР, PYTHON, PYGAME, 2048, ИГРА,  
КЛЕТОЧНОЕ ПОЛЕ

Предмет исследования – игра на клеточном поле.

Цель исследования – разработка игры 2048 для развлечения и  
улучшения стратегического мышления.

В процессе выполнения проекта ...

В результате выполнения проекта ...

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Описание игры	5
2 Проектирование игры	5
3 Реализация игры	5
ЗАКЛЮЧЕНИЕ	6
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	7
ПРИЛОЖЕНИЕ А	8

## ВВЕДЕНИЕ

Цель исследования – разработка игры для развлечения и улучшения стратегического мышления.

Для достижения поставленной цели требуется решить следующие задачи:

- 1) предложить идею для игры;
- 2) спроектировать приложение для игры;
- 3) реализовать спроектированное приложение.

## **1 Описание игры**

*Идея создания игры 2048 не нова однако до сих пор остаётся одной из лучших игр. Поэтому данный проект целиком и полностью посвящён данной игре.*

## **2 Проектирование игры**

*Основная суть игры путём нажатия 4 кнопок (вперёд, влево, вправо, назад) дойти от 2 до 2048. Механикой данной игры является удвоение числа то есть: при соприкосновении двух блоков с одинаковым значением остаётся один, а значение на блоках удваивается. Проигрыш происходит в том случае если на поле (стандартным полем является поле 4x4) закончатся пустые клетки (клетки на которых нет значений). Также сложность игры добавляет то, что цифры которые спамятся после каждого хода игрока выбираются рандомно между 2 и 4.*

## **3 Реализация игры**

*Для реализации проекта были задействованы библиотеки `pygame`, `sys`, `random` и среда разработки `PyCharm`.*

*Основные спроектированные механики это спам рандомных цифр при помощи библиотеки `random` и передвижение блоков.*

## ЗАКЛЮЧЕНИЕ

*Цель и идея проекта были реализованны в полной мере. Реализация основной идеи и механик была осуществлена.*

*Игра 2048 состоит из различных механик, но как основные можно выделить spawn новых объектов(блоков с цифрами) и их передвижение. Для обеих этих механик были спроектированы алгоритмы которые успешно работают.*

*При реализации проекта потребовалось воспользоваться библиотеки `pygame`, `sys`, `random` и среда разработки PyCharm.*

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 <https://lyceum.yandex.ru/>

2 <https://www.thepythoncode.com>

3

<http://juice-health.ru/raznoe/933-chto-takoe-2048-i-pochemu-eti-igry-tak-po-pulyarny>

## ПРИЛОЖЕНИЕ А

(обязательное)

### Листинг

```
def game():
    board = []
    for i in range(col_blocks):
        board += ["0"] * col_blocks
    new_num(board, 2)
    blocks = []

    for i in range(4):
        for j in range(4):
            blocks.append([pygame.Rect((i * 100) + 30, (j * 100) + 135, 90, 90), pygame.color.Color('white')])

    while True:
        for event in pygame.event.get():
            exit(event)
            if event.type == pygame.KEYDOWN:
                if event.key == pygame.K_w:
                    board = move_manager(board, "w")
                if event.key == pygame.K_s:
                    board = move_manager(board, "s")
                if event.key == pygame.K_a:
                    board = move_manager(board, "a")
                if event.key == pygame.K_d:
                    board = move_manager(board, "d")
        screen.fill(pygame.color.Color('white'))
        fon = pygame.transform.scale(load_image('bg.jpg'), (width, height))
        screen.blit(fon, (0, 0))
        name_a_game = font.render("2048", True, pygame.color.Color('orange'))
        screen.blit(name_a_game, (180, 50))
        for block in blocks:
            pygame.draw.rect(screen, block[1], block[0])
        for i in range(4):
            for j in range(4):
                screen.blit(text_builder(board, i, j)[0], text_builder(board, i, j)[1])
        if check_lose(board):
            gameOver()
        elif check_win(board):
            win()
        pygame.display.flip()
        clock.tick(fps)
```





