

# Breakout

Farukh Rustamov

19.05.2024

## 1 Úvod

Cílem bylo implementovat klasickou arkádovou hru Breakout. Během projektu jsem vytvořil hru s možností nastavení různých parametrů a realizoval několik úrovní obtížnosti.

## 2 Popis hry Breakout

Breakout je arkádová hra, ve které hráč ovládá platformu, která odráží míč, aby zničil všechny cihly na obrazovce. Hráč musí zabránit tomu, aby míč spadl mimo hrací pole, pohybem platformy doleva a doprava.

## 3 Realizace hry

### 3.1 Hlavní komponenty hry

- **Platforma:** ovládá se hráčem, pohybuje se doleva a doprava, odráží míč.
- **Míč:** odráží se od platformy, zdí a cihel. Obrázek míče může být přizpůsoben.
- **Cihly:** mají různou pevnost, která se určuje počtem zásahů potřebných k jejich zničení. Obrázky cihel mohou být přizpůsobeny.
- **Bonusy:** padají z rozbitých cihel a mohou hráči poskytnout různé vylepšení.

### 3.2 Herní objekty

- **Platforma:** realizována ve třídě `Paddle`. Ovládá se pomocí šipek.
- **Míč:** realizován ve třídě `Ball`. Mění směr při srážce s platformou, zdmi a cihlami.
- **Cihly:** realizovány ve třídě `Brick`. Mají různé úrovně pevnosti, které ovlivňují jejich barvu a obrázek.

- **Bonusy:** realizovány ve třídě `Bonus`. Padají z rozbitých cihel a aktivují vylepšení při sbírání.

## 4 Ukázka kódu

```
import pygame
import numpy as np
from settings import *

class Ball:
    def __init__(self, speed, radius=ball_radius, image_path=None):
        self.radius = radius
        self.rect = pygame.Rect(window_width // 2 - radius, window_height // 2 - radius, radius * 2, radius * 2)
        self.speed = np.array(speed) if isinstance(speed, list) else np.array([speed, -speed])
        self.image = pygame.image.load(image_path) if image_path else None
        if self.image:
            self.image = pygame.transform.scale(self.image, (radius * 2, radius * 2))

    def move(self):
        self.rect.x += self.speed[0]
        self.rect.y += self.speed[1]
        if self.rect.left <= 0 nebo self.rect.right >= window_width:
            self.speed[0] = -self.speed[0]
        if self.rect.top <= 0:
            self.speed[1] = -self.speed[1]

    def draw(self, surface):
        if self.image:
            surface.blit(self.image, self.rect.topleft)
        else:
            pygame.draw.ellipse(surface, self.color, self.rect)
```

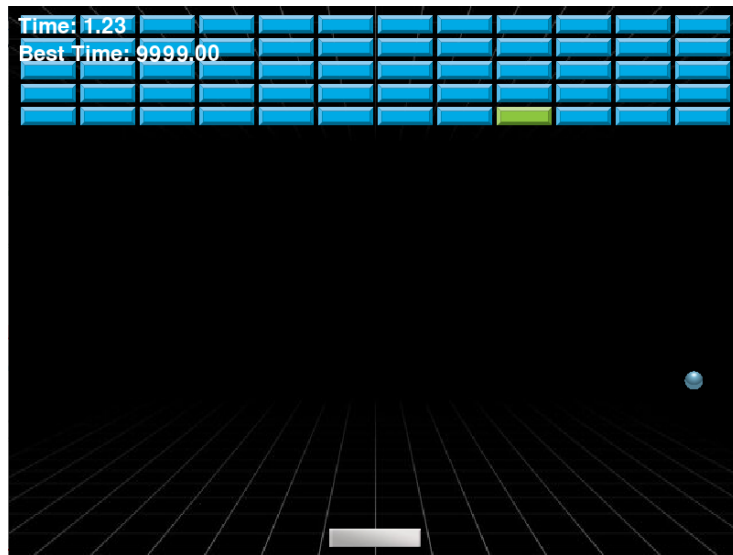
## 5 Hlavní moduly hry

### 5.1 Paddle (Platforma)

- Ovládá se hráčem pomocí šipek vlevo a vpravo.
- Zobrazuje obrázek platformy a může měnit velikost.

### 5.2 Ball (Míč)

- Pohybuje se po obrazovce, odráží se od platformy, zdí a cihel.
- Mění směr pohybu při srážce s objekty.



Obrázek 1: Herní proces ukazující platformu, míč a cihly

### 5.3 Brick (Cihla)

- Má úroveň pevnosti, které určují počet zásahů potřebných k jejich zničení.
- Zobrazuje různé obrázky v závislosti na úrovni pevnosti.

### 5.4 Bonus (Bonus)

- Vypadává z rozbitých cihel.
- Pohybuje se dolů po obrazovce a aktivuje různá vylepšení při sbírání.

## 6 Nastavení a parametry hry

Hra podporuje různé nastavení a parametry, které lze měnit v souboru `settings.py`.  
Příklad nastavení:

```
# Window settings
window_width = 800
window_height = 600
bg_color = (0, 0, 0)
fps = 60

# Paddle settings
paddle_width = 100
paddle_height = 20
```

```

paddle_speed = 10

# Ball settings
ball_radius = 10
ball_speed = [5, -5]

# Brick settings
brick_width = 60
brick_height = 20
brick_colors = {
    0: (128, 128, 128),
    1: (255, 0, 0),
    2: (255, 165, 0),
    3: (255, 255, 0),
    4: (0, 128, 0)
}
brick_rows = 5
brick_columns = 12

```

## 7 Funkce úrovně Kreační

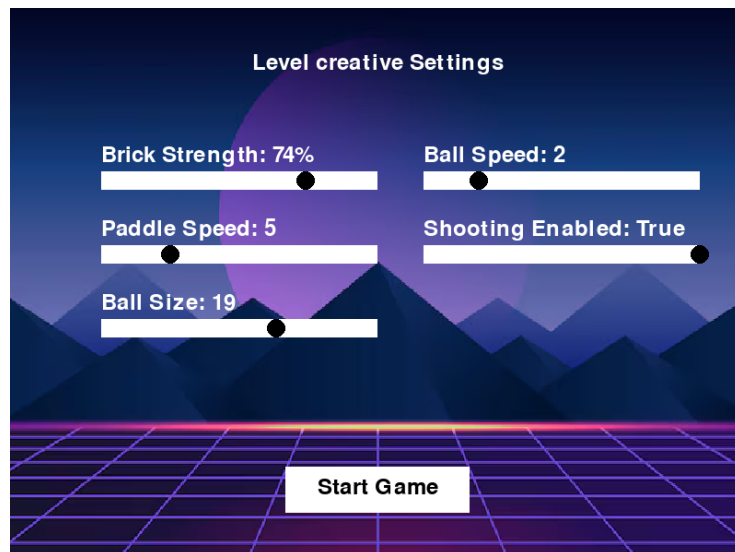
Jednou z hlavních funkcí hry je úroveň Kreační, kde lze plně přizpůsobit všechny parametry:

- Pevnost cihel
- Rychlost míče
- Přítomnost bonusů
- Možnost střelby
- Rychlost platformy
- Velikost míče

## 8 Hudba a zvukové efekty

Hra obsahuje hudební doprovod a zvukové efekty:

- Hudba na pozadí pro menu, kreační úroveň a hlavní herní proces.
- Zvukové efekty pro údery, ničení cihel, sbírání bonusů a další herní události.



Obrázek 2: Nastavení úrovně Kreační

## 9 Ukázka kódu pro spuštění hry

```
from menu import Menu

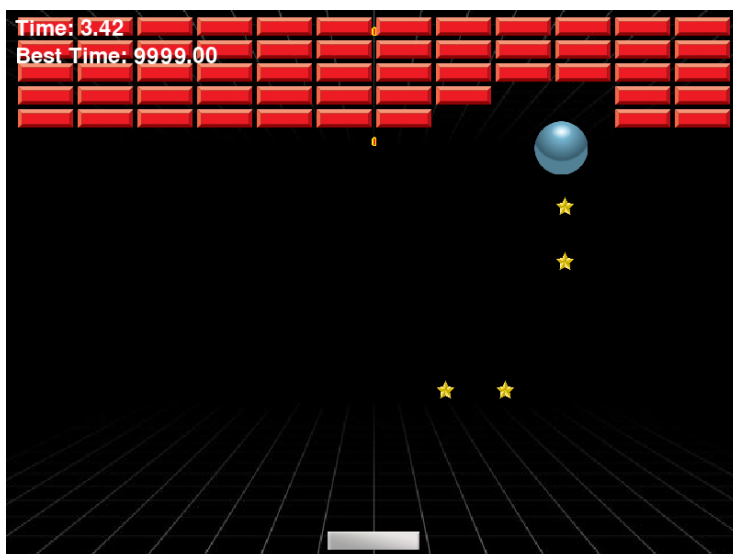
if __name__ == "__main__":
    menu = Menu()
    menu.run()
```

## 10 Závěr

Implementoval jsem hru Breakout s podporou nastavení různých parametrů a několika úrovněmi obtížnosti. Hra může být rozšířena přidáním nových úrovní, bonusů a dalších vylepšení. Tato realizace umožňuje hráčům užívat si klasickou arkádovou hru s možností přizpůsobení a různorodými herními prvky.

## 11 Další informace

- **Dokumentace:** projekt obsahuje podrobnou dokumentaci pro nastavení a používání hry.
- **Zdrojový kód:** zahrnuje všechny potřebné soubory pro spuštění hry.



## 12 Odkazy na zdroje

- **Zvuky a hudba:** Všechny zvuky a hudební soubory byly staženy z Free-sound a Incompetech.
- **Textury:** Textury a grafické prvky byly staženy z OpenGameArt.
- **Knihovny:** Hra byla vytvořena s použitím knihovny Pygame.

Tato zpráva pokrývá hlavní aspekty projektu, ukazující, jak lze hru Breakout použít pro zábavu a vytváření přizpůsobitelných herních úrovní.