

# **Dokumentace k zápočtovému programu**

## Exploding Atoms

# Obsah

1	Základní popis .....	1
2	Programátorská část .....	1
2.1	Použité datové struktury .....	1
2.1.1	Atom .....	1
2.1.2	Computer_player .....	1
2.1.3	Button .....	2
2.2	Fungování programu .....	2
2.3	Algoritmus .....	2
2.4	Použité moduly .....	3
3	Uživatelská část .....	3
3.1	Pravidla hry .....	3
3.2	Ovládání programu .....	3
4	Závěr .....	3

# 1 Základní popis

Zápočtový program Exploding Atoms je složený z této dokumentace a ze samotné hry Exploding Atoms, což je logická hra, jejíž pravidla jsou popsána v uživatelské části této dokumentace. Tato hra je primárně určena pro dva hráče, avšak jednoho z nich je možné nahradit počítačovým hráčem, tudíž je možno hrát i bez lidského protihráče.

Samotný program je ovládaný uživatelem prostřednictvím grafických prvků vytvořený za pomoci sady modulů pygame v jazyce Python, což však znemožňuje vytvoření rozumné testovací sady a tedy z tohoto důvodu tato sada není obsažena v tomto projektu.

## 2 Programátorská část

### 2.1 Použité datové struktury

#### 2.1.1 Atom

Každý z atomů hracího pole je vytvořený jako samotný objekt, avšak tyto objekty jsou samozřejmě propojené pomocí odkazů v datové struktuře listu (neighbours), kde jsou uloženy odkazy na sousední atomy. Tento list je doplněn až po vytvoření celého hracího pole plného danými atomy.

```
class Atom:
    def __init__(self, display, counter, text, neighbours, atoms, field_size):
        self.display = display
        self.neighbours = neighbours
        self.text = text
        self.counter = counter
        self.color = "black"
        self.atoms = atoms
        self.field_size = field_size
```

Obrázek 1: Objekt Atom

Kromě zmiňovaného listu sousedů obsahuje objekt Atomu ještě barvu, která je zobrazována v okně hry a která určuje vlastnictví atomu, a text, který určuje počet částic, které atom v současné chvíli obsahuje. Dále jsou objektu předány dvě datové struktury, a to objekt counter, pomocí kterého se určuje, kdo je právě na tahu a také slovník atoms v kterém je aktuální zastoupení barev v poli a pomocí kterého se určuje výherce celé hry a také ukončení samotné hry.

V této třídě také probíhají exploze a zvětšování počtu částic v daném atomu. Kromě toho funkce run() této třídy obstarává vykreslování grafické podoby atomu do okna hry.

#### 2.1.2 Computer\_player

Tato třída se stará o výběr nejvhodnějšího tahu při hře počítače. Nejdříve funkce této třídy byli založeny na vytváření minimaxu hry a vybírání nejlepšího možného tahu, avšak tato varianta byla velice pomalá i na malém poli 4x4, natož na poli 12x12 byli opravdu velké prodlevy. Proto nakonec tato třída sice používá základ minimaxu, avšak staví strom hry pouze pro tři následující kola (dvě kola odehraná počítačem a jedno kolo odehrané hráčem). Kvůli tomu je computer\_player dosti jednoduché porazit a jeho výhra je spíše výjimkou, ale za to je relativně rychlý.

Ještě předtím, než se vytvoří strom hry, je prohledáváno pole atomů a jsou hledány tzv. nebezpečné atomy, což jsou atomy, které mohou do určitého počtu kol explodovat. Nalezení takovýchto atomů

pomáhá k výraznému urychlení minimaxu v zásadě pomocí alfa-beta ořezávání, akorát se strom hry ořezá ještě před započítáním samotného algoritmu.

Jelikož se strom hry tvoří pouze pro tři následující kola a prohledávají se jen některé atomy (pouze ty nebezpečné), pro některé kombinace rozestavení hracího pole se nenajde ideální tah pomocí tohoto postupu. Proto pokud algoritmus minimaxu nenalezne atom, který bude v aktuálním kole zvolen, prochází se ještě nebezpečné atomy (atomy, které může počítač vybrat, takže ještě nejsou zabrány hráčem, a které vybuchnou do určitého počtu kol) a následně náhodně pomocí modelu random je vybrán atom z této množiny nebezpečných atomů.

### 2.1.3 Button

Všechna tlačítka mimo samotných atomů jsou vytvořena pomocí třídy Button, která jednak vykresluje tlačítka do oken a dále zachytává pomocí funkce `pygame.mouse.get_pressed()` případná jejich stisknutí (v tom případě zavolá funkci, která byla třídě předána). Tato třída vznikla hlavně kvůli relativně velkému počtu tlačítek v různých třídách, takže se díky ní neopakuje relativně obdobný kód v každé ze tříd.

## 2.2 Fungování programu

Jádrum samotné hry jsou třídy jednotlivých oken. Po spuštění programu je nejdříve vytvořena třída Menu, která vykreslí úvodní okno s tlačítkovým menu, v kterém lze nastavit možnosti hry a které přesměrovává uživatele do ostatních dvou oken.

Okno Náповědy je velice jednoduché, jelikož slouží pouze na zobrazení náповědy ke hře, a tedy tvoří takovou malou část uživatelské dokumentace. Dále toto okno umožňuje jen návrat do menu.

V neposlední řadě okno Hry starající se nepřekvapivě o běh samotné hry. Tato třída nejprve vytvoří herní pole o velikosti zvolené uživatelem a následně začne probíhat cyklus hry v kterém se zobrazuje celé herní pole a pomocí kontroly události (obstarána pomocí pygame) se detekují kliknutí myši na herní pole. Pokud uživatel klikne na nějaký atom zaznamená se poloha kliknutí myši, dohledá se atom, na který bylo kliknuto právě pomocí této polohy, a následně se zvýší počet částic tohoto atomu, přičemž může pochopitelně dojít k explozi atomu, což znamená, že se jako vlna začne zvětšovat počet částic v sousedních atomech, pokud tedy explodují i některé další atomy.

V průběhu cyklu se kontroluje konec hry za pomoci slovníku obsahujícího početní zastoupení barev na hrací ploše, tedy pokud se jednomu z hráčů povede obarvit celé herní pole v tomto slovníku bude jeho barva ohodnocena hodnotou stejnou jako je počet atomů v poli. Pokud je takto detekován konec hry ukončí se vlna explozí, jelikož jinak by se funkce zvětšování počtu částic v atomech zacyklila. Pokud je konec detekován zobrazí se výherce a kliknutím na obrazovku se uživatel vrátí do menu, kde může změnit nastavení a případně začít novou hru.

## 2.3 Algoritmus

V programu je použit v zásadě pouze základ minimaxu. Jeho výkonost je ještě vylepšena vyhledáváním potencionálně nebezpečných atomů, tedy takových atomů, které mohou do určitého počtu kroků teoreticky explodovat. Pomocí nich je zmenšen samotný strom hry, který minimax následně prochází. Avšak kvůli tomu, že při explozi může být změněna velká část herní plochy, je nutno při procházení všech potencionálních nebezpečných atomů tyto exploze v zásadě simulovat a následně do dalších kol předávat již změněnou plochu, aby byl minimax efektivní. Proto ještě před započítáním minimaxu program vytváří pole s aktuálními číselnými hodnotami všech atomů, které se poté předává a mění v jednotlivých uzlech stromu hry.

## 2.4 Použité moduly

V aplikaci je použita hlavně sada modulů pygame. Jen velice okrajově jsou použity ještě moduly os a random. Kdy OS modul používám pouze na skrytí uvítací hlášky od pygame a modul random na náhodné vybírání atomu, který bude zvolen z množiny potencionální možností při tahu počítače.

## 3 Uživatelská část

### 3.1 Pravidla hry

Pravidla celé hry jsou napsaná i v samotném programu, ale jsou v zásadě dosti jednoduchá. Hráči střídavě vybírají políčka neboli tzv. atomy z hracího plánu, jehož velikost je možné zvolit ze tří variant. Na počátku hry nepatří žádné atomové jádro nikomu a nikde nejsou žádné částice, tedy hrací pole je zaplněno samými nulami a všechny atomy mají základní černou barvu (znázornění toho, že nikomu nepatří). Při vybírání nelze zvolit atom, který je již zabrán soupeřem, tedy je obarvený soupeřovou barvou. Když je atom naplněn, tj. obsahuje dostatečný počet částic (v políčku se vyskytne tak velké číslo, jako má políčko přímých sousedů – tedy podle umístění políčka čísla 2 pro rohové atomy, 3 pro atomy na krajích a 4 pro vnitřní atomy), exploduje a rozdělí se mezi sousední atomy (atomy s kterými přímo sdílí stěnu, tedy se nejedná o diagonální směr přerozdělení). Při explozi případné soupeřovy částice, které byly v zasažených atomech, přebarví na svou barvu. Ostatní atomy zůstávají beze změny, ale pokud je nějaký atom znovu naplněn následuje další exploze. Vyhrává hráč, který eliminuje soupeře, tedy jinak řečeno dokáže obarvit celé hrací pole svou barvou.

### 3.2 Ovládání programu

Hlavní okno aplikace je sestaveno z dvou tlačítek (Start a Nápověda), která otevírají nová okna, dvou dalších tlačítek, která umožňují nastavit, buď “lidského” soupeře nebo nastaví jako soupeře počítač, a tři tlačítka, která umožňují nastavit velikost herního pole.

Dále program obsahuje dvě další již zmíněná okna, a to okno samotné hry a okno obsahující nápovědu ke hře. Obě tyto okna obsahují zpětná tlačítka, která umožňují návrat zpět do hlavního okna.

## 4 Závěr

Hru Exploding Atoms by šlo rozhodně vylepšit několika způsoby, ale podle mého jsou největší rezervy v třídě computer\_player, tedy v algoritmu, který se snaží nalézt nejlepší možný tah. Hlavně kvůli tomu, že já kvůli úspoře času vytvářím strom hry pouze na tři kola dopředu.

Samotný vývoj programu mi přišel zajímavý hlavně díky širokým možnostem modelu pygame. Co se týče nejtěžší části samotného projektu, tak to byla rozhodně implementace třídy computer\_player a algoritmu minimaxu společně s jeho zrychlováním.