



**STŘEDNÍ PRŮMYSLOVÁ  
ŠKOLA POLYTECHNICKÁ**  
**CENTRUM ODBORNÉ PŘÍPRAVY ZLÍN**  
Nad Ovčírnou IV 2528, 760 01 Zlín

Maturitní práce – Informační technologie

**Vývoj umělé inteligence pro konzolovou hru  
Námořní bitva**

Autor práce:

**Vojtěch Gistr, 4IT**

2023/2024

# Obsah

Úvod.....	4
1. Základní znalosti .....	5
1.1. Co je to umělá inteligence? .....	5
1.2. ASCII .....	6
1.3. Využití ASCII v konzolových aplikacích .....	7
1.4. Pravidla hry Námořní bitva .....	7
1.4.1. Příprava hry .....	8
1.4.2. Základní verze hry .....	8
1.4.3. Pokročilá verze hry .....	9
1.4.4. Verze s tužkou a papírem .....	9
2. Návod ovládání hry v konzoli.....	11
2.1. Zahájení hry .....	11
2.2. Rozložení lodí.....	11
2.3. Odstřelování .....	12
2.4. Konečná fáze hry .....	13
3. Mozek umělé inteligence .....	15
3.1. Nejběžnější patterny lidského myšlení .....	15
3.1.1. Umístění lodí.....	15
3.1.2. Odstřelování lodí.....	16
3.2. Patterny vyvinuté umělé inteligence .....	16
3.2.1. Umístění lodí.....	17
3.2.2. Odstřelování lodí.....	18
3.3. Možná vylepšení odstřelování AI .....	19
3.3.1. Hunt + Parity algoritmus.....	19
3.3.2. Funkce hustoty pravděpodobnosti .....	20
4. Uživatelské rozhraní a řídicí logika .....	23
4.1. Tvorba a vykreslování herní plochy.....	23
4.2. Detekce stisknutých kláves.....	24
4.3. Rozmístění lodí.....	24
4.4. Střelba.....	25
Závěr .....	26
Seznam použité literatury .....	<b>Error! Bookmark not defined.</b>

Seznam použitých symbolů a zkratk .....	28
Seznam obrázků .....	29

# ÚVOD

Tato maturitní práce vycházela z mého zájmu o řešení komplexnějších problémů a pochopení různých konceptů z matematiky, softwaru či jiných oblastí. Byla zvolena desková hra s názvem Námořní bitva, která se jevila jako dobrý kandidát díky své oblíbenosti a jednoduchým pravidlům. V mé realizaci si může hrát hru jen jeden hráč, sám proti umělé inteligenci, klidně i z pohodlí domova.

Cílem bylo vytvořit algoritmus v roli protihráče, a to dobrý tak, že jej bude těžké rozeznat od reálného hráče, tedy člověka. Toho jsem se snažil dosáhnout pomocí detailního pochopení pravidel hry a alespoň základního myšlenkového pochodu hráče. V tomto případě, kdy jsou pravidla relativně jednoduchá, je představivost málo omezená, což nabízí více možností, jak algoritmus vyvinout.

V rámci této práce jsem se zaměřil na analýzu pravidel Námořní bitvy, abych plně porozuměl dynamice hry. Dále jsem se pokusil vytvořit sofistikovaný algoritmus, který bude schopen strategicky rozmísťovat lodě a provádět útoky s cílem maximalizovat šance na vítězství. Pro hráče bude rozmísťování lodí samozřejmě možné i pomocí interaktivního a uživatelsky přívětivého rozhraní. V práci jsem použil vlastní obrázky z chystané aplikace.

# 1. ZÁKLADNÍ ZNALOSTI

V této části jsou vysvětleny základní termíny potřebné k pochopení daného tématu.

## 1.1. Co je to umělá inteligence?

V posledních desetiletích se objevila řada definic umělé inteligence (AI), ale John McCarthy v publikaci z roku 2007 nabízí následující definici: *„Je to věda a technika vytváření inteligentních strojů, zejména inteligentních počítačových programů. Souvisí s podobným úkolem, jakým je využití počítačů k pochopení lidské inteligence, ale AI se nemusí omezovat na metody, které jsou biologicky pozorovatelné.“* [1]

Nicméně desítky let před touto definicí byl zrod rozhovoru o umělé inteligenci označen zásadním dílem Alana Turinga *„Computing Machinery and Intelligence“* [2], které bylo publikováno v roce 1950. V této práci si Turing, často označovaný jako *otec informatiky*, klade následující otázku: *„Mohou stroje myslet?“* Na základě toho nabízí test, dnes známý jako *Turingův test*, v němž by se lidský tazatel pokusil rozlišit mezi počítačovou a lidskou textovou odpovědí. Ačkoli tento test prošel od svého zveřejnění mnoha zkoumáním, zůstává důležitou součástí historie umělé inteligence a také trvalým konceptem v rámci filozofie, neboť využívá myšlenky týkající se lingvistiky.

Stuart Russell a Peter Norvig pak vydali knihu *Artificial Intelligence: A Modern Approach* [3], která se stala jednou z předních učebnic v oblasti studia umělé inteligence. V ní se zabývají čtyřmi možnými cíli či definicemi umělé inteligence, které rozlišují počítačové systémy na základě racionality a myšlení vs. jednání:

### **Lidský přístup:**

- Systémy, které myslí jako lidé
- Systémy, které jednají jako lidé

### **Ideální přístup:**

- Systémy, které myslí racionálně
- Systémy, které jednají racionálně

Definice Alana Turinga by spadala do kategorie *„systémy, které jednají jako lidé“*.

Ve své nejjednodušší podobě je umělá inteligence obor, který kombinuje počítačovou vědu a robustní soubory dat, aby umožnil řešení problémů. Zahrnuje také podobory strojového učení a hlubokého učení, které jsou často zmiňovány ve spojení s umělou

inteligencí. Tyto obory se skládají z algoritmů umělé inteligence, které se snaží vytvářet expertní systémy, jež na základě vstupních dat provádějí předpovědi nebo klasifikace.

Umělá inteligence prošla v průběhu let mnoha cykly humbuků, ale i pro skeptiky se zdá, že vydání ChatGPT od OpenAI znamená zlom. Naposledy, když se generativní umělá inteligence takhle rozmohla, došlo k průlomů v oblasti počítačového vidění, ale nyní se skok kupředu odehrává v oblasti zpracování přirozeného jazyka. A nejde jen o jazyk. Generativní modely se mohou učit také gramatiku softwarového kódu, molekul, přírodních obrazů a řady dalších typů dat. [\[4\]](#)

## 1.2. ASCII

ASCII, standardní kódovací formát pro elektronickou komunikaci mezi počítači, vznikl s cílem řešit nedostatečnou interoperabilitu mezi různými počítačovými systémy před jeho zavedením. Každý výrobce počítače měl vlastní způsob reprezentace znaků, což způsobovalo problémy v komunikaci. V roce 1961 Bob Bemer předložil návrh společného počítačového kódu Americkému národnímu standardizačnímu institutu (ANSI), což vedlo k schválení ASCII jako amerického standardu dne 17. června 1963. [\[5\]](#)

Navzdory schválení ASCII na počátku čelilo omezené akceptaci, protože společnost IBM zvolila přednostně EBCDIC ve svých počítačích. Nicméně ASCII podstoupilo následné revize v letech 1965, 1967, 1968, 1977 a 1986. Dne 11. března 1968 nařídil prezident Spojených států Lyndon B. Johnson, aby bylo ASCII označeno za federální standard k minimalizaci nekompatibility v počítačových a telekomunikačních systémech federální vlády. Dále nařídil, že všechny nové počítače a přidružené zařízení zakoupené vládou USA od 1. července 1969 musí být kompatibilní s ASCII. [\[5\]](#)

Původně navržené pro telepřístroje, ASCII našlo široké uplatnění v osobních počítačích, zejména s uvedením prvního PC od společnosti IBM v roce 1981. ASCII využívá sedmimístná binární čísla, což umožňuje reprezentaci 128 různých znaků. Binární sekvence, jako například 1010000 pro velké písmeno P, slouží jako základ pro reprezentaci znaků. Přejchod na digitální počítače s osmibitovými byty zvýšil kapacitu ASCII na 256 znaků, vytvářejíc tak rozšířené ASCII, které bylo představeno společností IBM v roce 1981. Tato expanze zahrnovala řídicí příkazy, číslice, interpunkční znaménka a jak velká, tak malá písmena. [\[5\]](#)

Avšak omezení rozšířeného ASCII ve schopnosti podporovat všechny psané jazyky vedlo k vytvoření nových standardů, jako jsou Unicode a UCS. Unicode, představený v roce 1991, zahrnuje prvních 128 kódových kombinací ASCII, což ho činí zpětně kompatibilním a zároveň schopným zastřešit širší škálu znaků. Unicode zaznamenal rozsáhlou adopci, zejména na World Wide Web, a stal se převládajícím systémem kódování znaků na začátku 21. století. [\[5\]](#)

### 1.3. Využití ASCII v konzolových aplikacích

Do textové konzolové aplikace lze přidat vizuální design, i když není možno používat grafické soubory. V konzolové aplikaci se často používají alfanumerické znaky k vytváření prvků designu, tedy **ASCII design**.

Při správném použití může ASCII udělat rozhraní zajímavějším. ASCII je nutno pečlivě vyvažovat, aby hráče nezahlcovalo a neodvádělo pozornost od důležitého textu. [\[6\]](#)

Možnosti, jak vytvořit originální design:

- **Generátory obrázků a textů**, které vytvářejí obrázky nebo víceřádková slova pomocí specifických nástrojů.
- **Konvertory**, a to stažení nebo použití online konvertoru k převodu vytvořené grafiky na alfanumerické znaky.
- **Vytváření návrhů** po jednom alfanumerickém písmenu nebo symbolu.

### 1.4. Pravidla hry Námořní bitva

Námořní bitva je válečná desková hra pro dva hráče, ve které se soupeři snaží uhodnout umístění lodí protivníka a potopit je. Verze hry s papírem a tužkou sahá až do první světové války, ale většina lidí zná hru přes plastovou verzi, kterou poprvé uvedla společnost Milton Bradley v roce 1967. Od té doby vzniklo několik videoher a variant pro chytré telefony. Dnes je desková verze hry vyráběna společností Hasbro, která Milton Bradley získala v roce 1984. [\[7\]](#)

Mechanika hry je jednoduchá. Každý hráč skrývá lodě na plastové mřížce obsahující svislé a vodorovné souřadnice. Hráči se střídají ve volání souřadnic řádku a sloupce na mřížce druhého hráče s cílem identifikovat čtverec obsahující loď.

Herní deska, kterou každý hráč dostane, má dvě mřížky horní a dolní. Dolní mřížka slouží hráči ke skrytí polohy svých vlastních lodí, zatímco horní mřížka slouží k zaznamenání výstřelů směřovaných k protivníkovi a k zaznamenání toho, zda tyto výstřely byly zásahy nebo minuty.

### **1.4.1.Příprava hry**

Každý hráč obdrží na začátku hry jednu herní desku, pět lodí různých délek a dva typy kolíků, bílé a červené. Každá loď má otvory, do kterých jsou vkládány červené kolíky. Ty slouží jako označení zasažených částí lodi. Bílé kolíky slouží pro označení nezasazených polí.

Hráč má k dispozici pět lodí:

- Letadlová loď s pěti otvory
- Bitevní loď s čtyřmi otvory
- Křižník s třemi otvory
- Ponorka s třemi otvory
- Torpédoborec se dvěma otvory

Oba hráči by měli být usazeni tak, aby na sebe směřovali přes herní stůl. Jejich cílové mřížky jsou vertikálně umístěny tak, aby žádný hráč nemohl vidět oceánskou mřížku a umístění lodí svého protivníka.

Před začátkem hry každý soupeř tajně umístí svých pět vlastních lodí na oceánskou mřížku (umístěnou v dolní části desky), rozmístěním lodí a ukotvením kolíků do otvorů na mřížce. Každá loď musí být umístěna horizontálně nebo vertikálně napříč mřížkovými prostory – diagonální umístění není povoleno. Lodě nesmějí vyčnívat za hranice mřížky. Lodě se mohou dotýkat, ale nesmí obsadit stejný prostor na mřížce. Po zahájení hry není povoleno měnit pozici lodí.

### **1.4.2.Základní verze hry**

Hráči se střídají v odpalování střel voláním souřadnic na mřížce s cílem zasáhnout loď svého protivníka. Ve svém tahu vysloví písmeno a číslo identifikující řádek a sloupec na cílové mřížce protihráče. Protivník poté zkontroluje tuto souřadnici na své oceánské mřížce a verbálně odpoví, zda střela minula či zasáhla loď.



Každý svůj výstřel nebo pokus o palbu na nepřítele zaznamenávají hráči na své cílové mřížce umístěné v horní části desky. Používají bílé kolíky k označení minulých pokusů a červené kolíky k zaznamenání zásahů. S postupem hry budou červené kolíky postupně odhalovat velikost a umístění lodí protivníka.

Jakmile jsou všechny otvory na jedné lodi vyplněny červenými kolíky, oznámí hráč protihráči, že loď je potopena. V klasické hře se tato fáze vyjadřuje slovy: „*Potopil jsi mou bitevní loď!*“ První hráč, který úspěšně potopí všech pět lodí svého protivníka, vyhrává hru.

### 1.4.3. Pokročilá verze hry

Zkušení hráči Námořní bitvy často volí upravenou verzi hry nazývanou variace *Salvo*. Základní pravidla zůstávají stejná s následujícími výjimkami:

- 1) V prvním kole hry nevyvoláváte jednotlivé tipy, ale najednou oznámíte pět střel a každou z nich označíte bílým kolíkem na své cílové mřížce.
- 2) Poté, co dokončíte salvo pěti střel, váš protivník sdělí, které střely byly úspěšné zásahy a které lodě zasáhl.
- 3) Za každý zásah na své cílové mřížce nahradíte bílé kolíky červenými. Zároveň váš protivník vloží červené kolíky do odpovídajících otvorů na eventuálně zasažených lodích.
- 4) Hráči se střídají tímto způsobem, dokud se jedna z jejich lodí nepotopí. V tom okamžiku ztratí jeden pokus ze svého salva. Například, pokud potopíte jednu ze svých lodí, váš salvo se nyní sníží na čtyři střely; pokud potopíte dvě lodě, salvo bude tři střely, a tak dále.
- 5) Pokračujte ve hře, dokud jeden hráč úspěšně nepotopí všechny lodi protivníka a nevyhraje hru.

### 1.4.4. Verze s tužkou a papírem

Hru je možné hrát také klasickým způsobem s papírem a tužkou. Lze si vyrobit vlastní herní desku tak, že nakreslíte oceánské mřížky na papír a označíte horizontální a vertikální řady k vytvoření souřadnic. Například pro základní oceán s 100 poli jsou označeny horizontální řady 1 až 10 a vertikální řady A až J.

Stejně jako u tradiční deskové hry, každý hráč bude potřebovat dvě oceánské mřížky. Jedna mřížka označuje skrytou pozici pěti lodí a jakékoliv zásahy, které protivník provede na těchto lodích. Druhá mřížka sleduje zasažené a minuté střely, které provede hráč při střelbě na protivníka. Když vystřelí a zasáhne, označí odpovídající čtverec symbolem „X“. Pokud mine, zaznamená to nulou, tečkou, nebo jiným znakem.

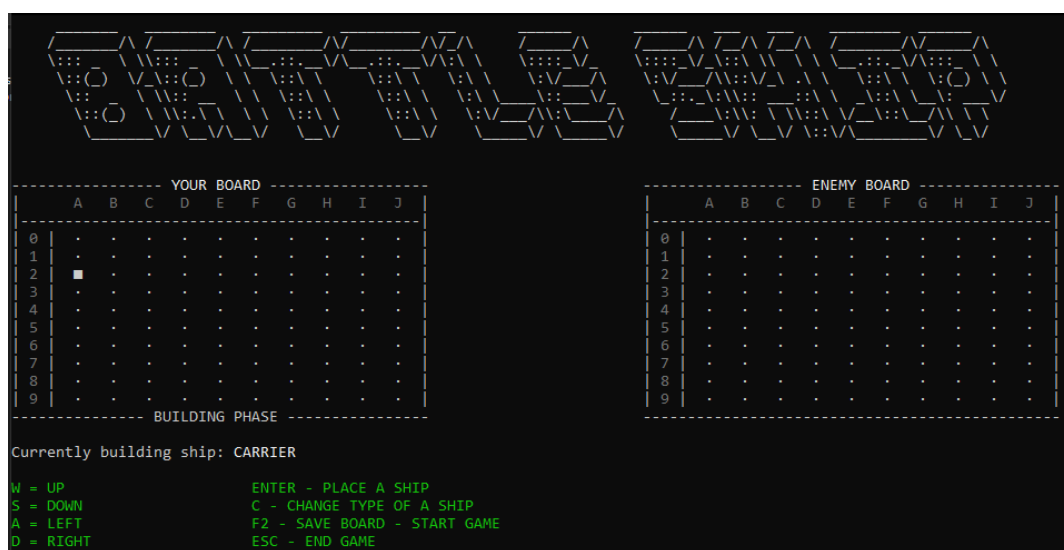
Pro obtížnější hru si hráči mohou vytvořit větší mřížku s více poli, kde použijí písmena A až Z pro označení vertikálních souřadnic a čísla 1 až 26 pro horizontální řady. Čím větší je oceánská mřížka, tím obtížnější je hra.

## 2. NÁVOD OVLÁDÁNÍ HRY V KONZOLI

V této verzi Námořní bitvy hraje hráč proti AI. Rozbor tohoto algoritmu se nachází v samotné práci.

### 2.1. Zahájení hry

Při otevření spouštěcího souboru se uživateli zobrazí herní plocha společně s ovládáním hry (zelený text, obrázek 1). Plocha je rozdělena na levé pole, které se chová jako dolní mřížka, a pravé pole, tedy horní mřížka.



Obrázek č. 1 - Uživatelské rozhraní pro konzolovou hru

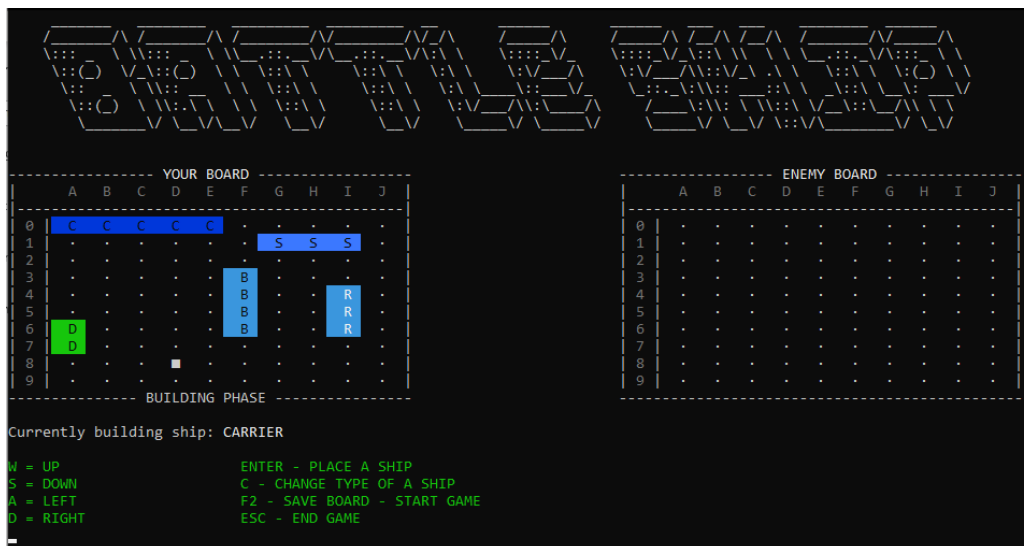
Hra je rozdělena na 3 fáze, která je vyobrazena v dolní části levého pole (obrázek 1). Tyto fáze jsou:

- „BUILDING PHASE“ – začáteční fáze, kdy hráč i soupeř umísťují své lodě na herní plochu.
- „SHOOTING PHASE“ – fáze hry, kdy na sebe hráč a soupeř střílí, tedy kdy reálná hra probíhá.
- „END PHASE“ – konec hry, kdy se vypíše vítěz na obrazovce.

### 2.2. Rozložení lodí

Pro započetí hry je nutno rozložit lodě, ty jsou reprezentovány písmeny a barvami. Splnění tohoto úkolu se dá dosáhnout pomocí bílého kurzoru tvaru čtverce, se kterým se

lze pohybovat pouze po poli hráče, tedy pole označené „*YOUR BOARD*.“ Pohyb s kurzorem zajišťují klávesy WASD, což je standardem v herním průmyslu. Ten byl poprvé použit v roce 1986 ve hře Dark Castle a po 10 letech ho adaptovala, dodnes známá, hra Half-Life.

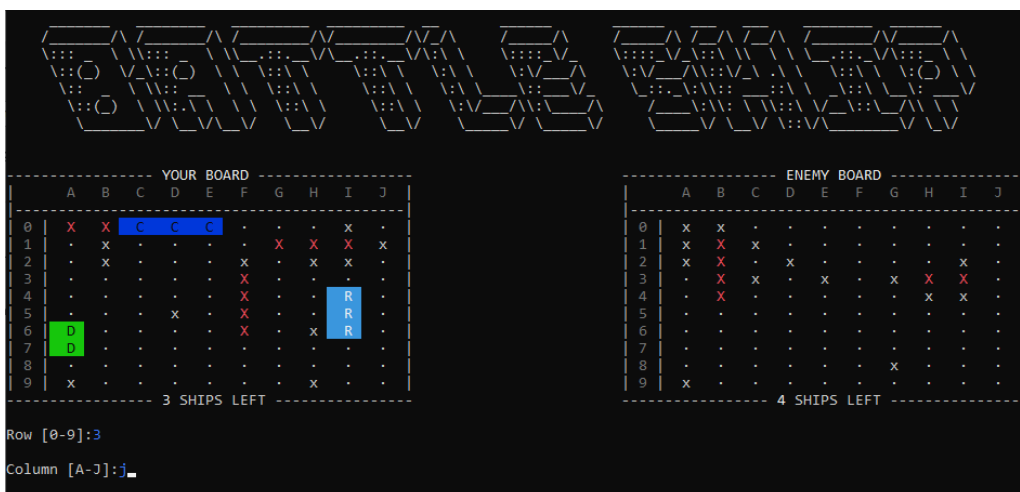


Obrázek č. 2 - Příklad fáze rozložení lodí

Pro položení lodě je nutno stisknout klávesu „*Enter*“. Slouží jako potvrzení pro položení právě označené lodě, ta je vyobrazena nad návodem k ovládání textem „*Currently building ship: CARRIER*“ (obrázek 2). Na budování i jiné fáze hry platí stejná pravidla jako uvedená v této práci. Tedy jednoduše je nelze stavět náhodně, či diagonálně. Pokud se hráč splete nebo změní názor kam by loď chtěl umístit, může ji odstranit pomocí klávesy „*Backspace*“. Pro změnu typu lodě je nutno stisknout klávesu „*C*.“ Hra se dá vždy ukončit pomocí klávesy „*ESC*.“

## 2.3. Odstřelování

Po stisknutí klávesy „*F2*“, pokud jsou lodě hráče rozmístěny, hra přechází do fáze „*SHOOTING PHASE*.“ Cíl hráče je sestřelit všechny lodě protihráče. Trefené lodě nebo její části jsou vyobrazeny červeným „*X*“, naopak minuté pole jsou šedým „*X*“ (obrázek 3).



Obrázek č. 3 – Příklad odstřelovací fáze hry

Poslání střely na protihráče je možno zadáním souřadnic níže pod herním polem, které přijímají písmena A-J a čísla 0-9. Písmena i čísla lze prohodit nebo je napsat dohromady pro rychlejší střely. Příkladem může být zadání souřadnic „0a,“ tento příkaz vystřelí přesně do levého horního rohu na pole protihráče a poté se zobrazí, jestli hráč trefil loď nebo ne.

V této části hry vidíme tam, kde byla předchozí fáze hry vypsána, počet zbývajících lodí, patřící k poli nad textem. Například v obrázku výše vidíme, že hráči zbývají pouze tři lodě a nepříteli čtyři. Zároveň je hráč také pod útokem umělé inteligence, což znamená, že algoritmus našel loď svého protihráče a pomalu ji odstřeluje. Jakmile se tohle stane, počet lodí hráče („YOUR BOARD“) klesne a bude mít pouze 2 lodě.

## 2.4. Konečná fáze hry

Jakmile jeden z hráčů odstřelí všechny lodě svého protihráče, hra se automaticky ukončuje a program vyhodnocuje výsledek. Podle pravidel, ten, kdo má všechny lodě zničené, prohrává.

Po ukončení fáze odstřelování, kdy si hráči navzájem ničí lodě, zobrazí se velký ASCII text s obsahem „Enemy wins,“ nebo „Player wins.“ Pokud se objeví „Enemy,“ vyhrává umělá inteligence, jestliže „Player,“ vyhrává hráč. Jsou také prozrazeny lokace nepotopených částí lodí hráče či soupeře, v závislosti na tom, kdo vyhrál (obrázek 4).

<pre> \$\$\$\$\$\$\$\$\ \$\$ \$\$    \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\ \$\$\$\$\$\  \$\$  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\ \$\$     \$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$ \$\$     \$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$ \$\$\$\$\$\$\$\$\  \$\$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$  \$\$\$ \   \$\$\$\  \$\$\$\           \\$\$\$\$\$\$\$ </pre>										
<pre>           \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\ </pre>										
<pre> ----- YOUR BOARD -----    A  B  C  D  E  F  G  H  I  J       0     X  X  X  X  X  .  .  .  X  X       1     .  X  .  .  .  .  X  X  X  X       2     X  X  .  X  .  X  .  X  X  .       3     .  .  .  X  X  X  .  .  X  .       4     .  X  .  .  X  X  .  .  X  .       5     X  .  .  X  .  X  .  .  X  .       6     X  .  X  .  X  X  .  X  X  .       7     X  X  .  X  .  .  X  .  .  .       8     X  .  .  .  .  X  .  .  .  .       9     X  .  .  .  .  X  .  X  .  .     ----- 0 SHIPS LEFT -----  </pre>										
<pre>           \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\           \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\  \$\$\$\ </pre>										
<pre> ----- ENEMY BOARD -----    A  B  C  D  E  F  G  H  I  J       0     X  X  .  .  .  .  .  .  .  .       1     X  X  X  .  .  .  .  .  .  .       2     X  X  .  X  X  X  X  .  .  .       3     X  X  X  X  X  X  X  X  X  X       4     .  X  X  X  X  X  X  X  X  X       5     .  .  .  .  .  .  C  .  X  .       6     .  .  .  X  X  C  X  X  X  .       7     X  .  .  .  .  C  .  X  X  X       8     .  S  S  S  .  .  .  X  .  X       9     X  .  .  .  X  .  .  .  X  .     ----- 2 SHIPS LEFT -----  </pre>										

Obrázek č. 4 – Příklad konce hry

### 3. MOZEK UMĚLÉ INTELIGENCE

Umělá inteligence (dále jen „AI“), jenž hraje za protivníka, je vyvinuta tak, aby byla co nejblíže myšlení člověka hrající tuto hru. V této části jsou tedy vysvětleny algoritmy, které řídí rozhodování AI ve všech třech fázích hry. To zahrnuje jak strategické umístění lodí, paměť, strategii střílení nepřátel.

#### 3.1. Nejběžnější patterny lidského myšlení

Myšlení každého jednotlivce je unikátní, avšak v rámci kolektivu existuje potenciál odhalit určité vzory nebo strategie, které jsou sdíleny větší skupinou lidí. Představme si, že tisíc jednotlivců je vzato v úvahu a jejich přístup k různým situacím, jako je umisťování lodí nebo taktika střílení, je zkoumán. I přes individuální rozdíly jsou identifikována opakující se chování.

Tato analýza může být využita k odhalení obecných vzorů neboli pattern, které mohou být dále využity pro hlubší pochopení lidského chování. Získané poznatky mohou být následně aplikovány v různých kontextech, od herních strategií po vědecký výzkum chování a rozhodování jednotlivců. Takový přístup umožňuje zkoumat podobnosti a odlišnosti v myšlení, což může přispět k lepšímu pochopení lidské psychologie a chování ve skupinových situacích.

##### 3.1.1. Umístění lodí

V kontextu umisťování lodí, patterny myšlení se mohou projevovat různými způsoby. Když jednotlivci přistupují k této strategické hře, mohou vyvinout určité postupy nebo taktiky založené na svých osobních preferencích, zkušenostech a analytických schopnostech.

Někteří hráči se mohou zaměřit na vytváření souvislých bloků lodí, což může být strategie zaměřená na zabránění soupeři v identifikaci a potopení jejich lodí. Jiní se mohou soustředit na rozprostření svých lodí po celé herní ploše, což může ztížit úkol soupeři při hledání a zasažení všech lodí.

Vzájemné kombinace lodí, jako například vytvoření „klíče“ pro soupeře, mohou být také součástí strategického přemýšlení. Někteří hráči preferují agresivnější přístup a

jdou rovnou po velkých a důležitých částech herní plochy, zatímco jiní volí opatrnější postup a snaží se postupně odhalit polohu všech lodí soupeře.

Významným faktorem může být i schopnost adaptace k chování soupeře, kdy hráč může upravovat svou strategii na základě toho, jak soupeř reaguje na jeho předchozí tahy.

### **3.1.2.Odstřelování lodí**

V kontextu strategií střílení se vzory myšlení hlavně týkají efektivního odhalování polohy lodí soupeře, tedy maximalizace šance na zásahy.

Někteří hráči preferují systematický přístup, kde postupně pokrývají celou herní plochu, aby zajistili, že žádná část nezůstane neprozkoumaná. To může být užitečné pro odhalení menších lodí a minimalizaci šance na únik. Ovšem, tato strategie může vyžadovat více tahů než když se hráč zaměřuje na konkrétní oblasti.

Naopak, jiní hráči mohou preferovat více agresivní přístup a cíleně útočit na oblasti, kde mají podezření, že se nachází větší loď. Tato taktika může zahrnovat sledování vzorů na herní ploše a snažení se předpokládat, kde by soupeř mohl umístit své loď s ohledem na dostupné informace.

Adaptabilita je klíčovým prvkem při střílení. Hráči by měli být schopni reagovat na změny v chování soupeře a přizpůsobit svou strategii podle nových informací. To může zahrnovat přehodnocení původních předpokladů o poloze lodí a optimalizaci střeleckých vzorů na základě nově získaných dat.

Celkově lze vzory myšlení při střílení v této hře charakterizovat kombinací systematického zkoumání, strategického útoku a schopnosti rychlé adaptace k nepředvídatelným situacím.

## **3.2. Patterny vyvinuté umělé inteligence**

Pro to, aby bylo docíleno co největší blízkosti této inteligence k člověku, je nutno využít i lidské chování jako takové. Pokud by tato nejzákladnější vlastnost algoritmu chyběla, nebylo by možno jej vyvinout legitimně. Příkladem je paměť a zrak. Jestliže si program nedokáže zapamatovat, kde naposledy střelil či jakým směrem loď směřuje,



sice to člověk může být, ale ne průměrný. Kdyby viděl loď protihráče, platilo by to stejné.

Je klíčové zdůraznit, že při této hře není program schopen uchovat informace mezi jednotlivými hrami, což vytváří výrazný rozdíl ve schopnostech adaptace mezi lidským hráčem a umělou inteligencí. Pro dosažení srovnatelných strategických dovedností by program musel mít přístup k ukládání a učení z předchozích her, buď v lokální databázi nebo v cloudu.

### 3.2.1. Umístění lodí

Tento algoritmus je vyvinut v této oblasti čistě náhodně. Program si na herní ploše vybere náhodný sloupec, řádek, směr, a to při každém položení jiného typu loď. Ihned po zvolení souřadnic je zkontrolováno, či loď lze opravdu položit. To záleží na několika faktorech:

- **Nelze**, uvedená data loď přesahují hrací plochu v pravé či levé části dvoudimenzionálního listu dat.
- **Nelze**, uvedená data loď přesahují hrací plochu v horní či spodní části dvoudimenzionálního listu dat.
- **Nelze**, uvedená data loď zasahují do jiné lodě.
- **Lze**, všechny výše zmíněné podmínky neplatí.

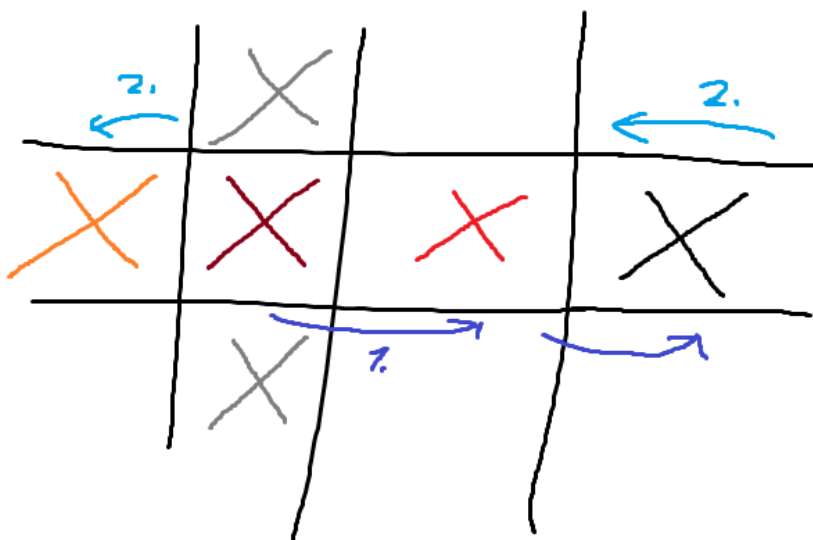
Je-li pozice loď vyhodnocena jako validní, je položena na hrací plochu, naopak jsou vybrány nová data a celá operace se opakuje.

Rozdíly mezi člověkem a programem v této části jsou evidentní a zásadní. Klíčovým faktorem jsou zkušenosti a strategie, které člověk může vytvářet a upravovat v průběhu hry. Na rozdíl od programu, který si pamatuje informace pouze do konce aktuální hry a začíná s čistým štítem při každém novém odstartování. Hráč má schopnost vytvářet a měnit své taktiky na základě nashromážděných zkušeností, což je něco, co program bez přístupu k lokální databázi, nebo cloudu nedokáže.

Lodě AI jsou umístěny ihned po tom, co hráč stiskne klávesu F2, sloužící pro start hry. V tomto případě program přejde do následující fáze hry, odstřelování lodí.

### 3.2.2. Odstřelování lodí

Na rozdíl od umísťování lodí se tato část myšlení neřídí pouhou náhodou. Sice prvotní střílení je náhodné, ale jakmile je sestřelena jedna část lodě, AI přechází do módu ničení nalezené lodě. Tento mód řídí specifický pattern, která je vysvětlena níže (obrázek 5)



Obrázek č. 5 – Příklad logiky odstřelení lodě umělou inteligencí

Prvním cílem AI je zjistit směr zasáhnuté lodě. Toho docílí tak, že si vezme poslední souřadnici, tedy tu zasáhnutou, a od ní přičítá či odčítá jedničku. Nejdříve si vezme řádek, a pokud je to možné, odečte a přičte číslo jedna. Z toho vzniknou dvě nové souřadnice, na které v jeho následujících tazích vystřelí. Pokud ani jedna ze souřadnic nebyla část lodě, stejnou operaci vykoná v rámci sloupců. Směr lodě je následně zaznamenán do paměti ve formátu „*HORIZONTAL\_LEFT*“, „*HORIZONTAL\_RIGHT*“, „*VERTICAL\_DOWN*“, nebo „*VERTICAL\_UP*“.

Pokud se stane, že loď nebyla stále sestřelena, což lze poznat podle počtu lodí ve spodní části pole, a střela v současnosti nezasáhla loď, směr v paměti se obrátí a v dalších tazích je stříleno vedle již zasáhnutých částí. Příkladem by byl směr „*VERTICAL\_DOWN*“, kdy AI směřuje své tahy vertikálně dolů. Pokud je pole najednou něco jiného než loď, další tahy jsou směřovány vertikálně nahoru, přičemž již střelené pole jsou přeskočena.

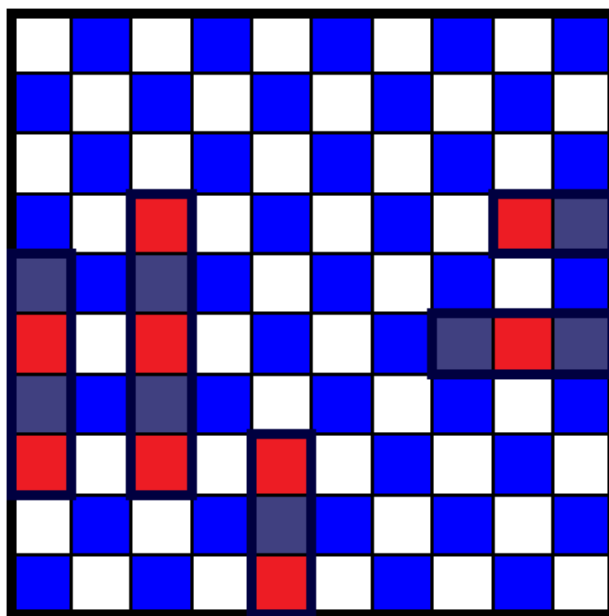
### 3.3. Možná vylepšení odstřelování AI

Mnoho informací je bráno z blogu DataGenetics [\[8\]](#), na kterém Nick Berry píše o analýze dat a algoritmech.

Nick Berry vystudoval obor raketového inženýrství a konstrukce letadel a získal magisterský titul v oboru leteckého a astronautického inženýrství. Po ukončení studia se připojil ke skupině přátel a založil softwarovou společnost, která se specializovala na elektronické mapování a plánování tras.

#### 3.3.1. Hunt + Parity algoritmus

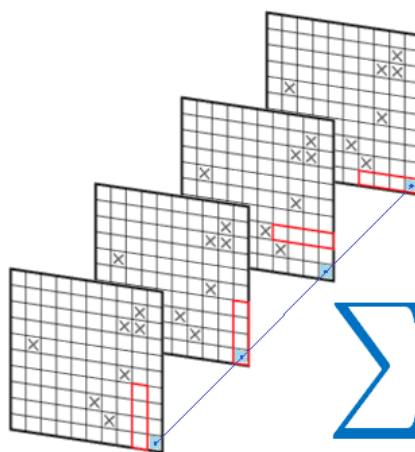
Tento princip instruuje algoritmus, aby bral v úvahu skutečnost, že lodě vždy zabírají liché i sudé pozice na herní ploše. Tuto informaci lze využít k vytvoření inteligentnějších strategií střelby. Například, na obrázku níže by se měl hráč vyhýbat střelbě na bílou destičku vlevo nahoře, pokud již v minulosti střílel na její modré sousedy. Možná si intuitivně v minulosti všiml a využil této taktiky při střílení na pole v šachovnicovém uspořádání. Začleněním principu parity do algoritmu se podle Nickových simulací podařilo zlepšit mediánovou rychlost dokončení jejich strategie na přibližně 62 tahů.



Obrázek č. 6 – Princip šachovnice [\[8\]](#)

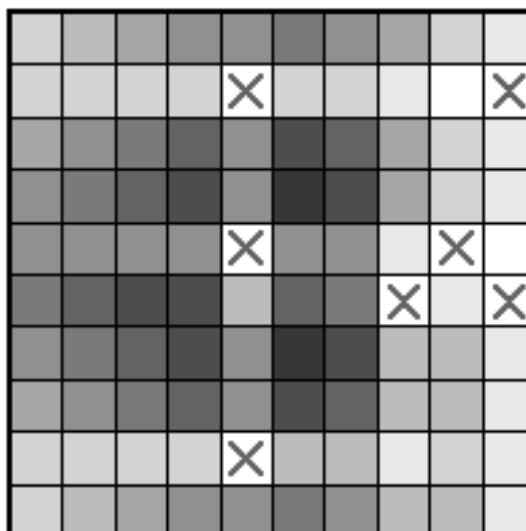
### 3.3.2. Funkce hustoty pravděpodobnosti

Nickův konečný navržený algoritmus nyní představuje značně zvýšenou výpočetní náročnost. Využívá funkce hustoty pravděpodobnosti, která umožňuje výpočet všech možných míst, kam by se mohla vejít každá zbývající loď na začátku každého tahu. S přítomností pěti lodí vzniká obrovské množství různých kombinací, které jsou následně sečteny. Každému políčku na hrací ploše je přiřazena pravděpodobnost obsahu části lodi s ohledem na již odkryté pole.



Obrázek č. 7 – Výpočet pravděpodobnosti, že dlaždice obsahuje loď, na základě všech možných rozložení hrací plochy [8]

V samém začátku hry, kdy nejsou žádná pole odkryta, mají všechna políčka přibližně stejnou pravděpodobnost obsahu lodě. Nicméně s každým novým výstřelem se pravděpodobnost mění. Na některých místech se stává nemožným, na jiných téměř jistým. Ilustrativně lze tuto změnu pozorovat na přiloženém obrázku, kde sedm nezasažených políček X a tmavší políčka reflektují relativně vysokou pravděpodobnost obsahu části lodi.

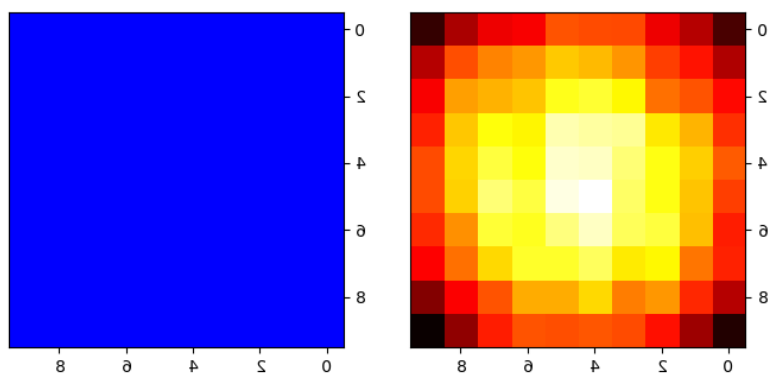


Obrázek č. 8 – Příklad distribuce se sedmi ranami na hracím poli [8]

Nick provedl simulaci 100 milionů her pro tento pravděpodobnostní přístup a srovnání s předchozími algoritmy. Graf níže shrnuje výsledky a zdůrazňuje, že tento nový pravděpodobnostní algoritmus výrazně překonává jednodušší přístupy. Zajímavě ukazuje, že dokáže dohrát 50 % her během průměrně ~42 tahů, což je v porovnání s jinými metodami velký pokrok. Tento algoritmus je tak v oblasti deskových her považován za výjimečně silný.

Uživatel Redditu s přezdívkou DataSnaek přetvořil tento pravděpodobnostní algoritmus do programovacího jazyka Python a jeho vnitřní výpočty vizualizoval v poutavém GIFu (obrázek 9). Na levé straně níže lze pozorovat pravděpodobnost, že každý čtverec na herní ploše obsahuje část lodi. Čím jasnější je barva (od bílé přes žlutou, červenou až po černou), tím je pravděpodobnější, že se na daném místě nachází loď. Je bráno v úvahu, že lodě mohou zabírat více po sobě jdoucích políček.

Vpravo zobrazuje algoritmus, který v každém tahu vystřelí na místo s nejvyšší pravděpodobností obsahu lodní části. Modrá barva reprezentuje neznámé oblasti, červená označuje chybně střelená místa, hnědá indikuje potopené lodě, a světle modré značí zasažené, ale nepotopené, lodě. Tato vizualizace ukazuje postupné vylepšování algoritmu v průběhu hry na základě aktualizovaných pravděpodobností a efektivní volby míst pro výstřely.



Obrázek č. 9 – Matice pravděpodobnosti jako tepelná mapa pro každé pole po každém tahu ve hře [\[9\]](#)

## 4. UŽIVATELSKÉ ROZHRAŇÍ A ŘÍDÍCÍ LOGIKA

V této kapitole je vysvětleno, jakým způsobem jsou zobrazována data uživateli. To může znamenat velký text „*BATTLESHIP*,“ jenž je k nalezení v části návodu k ovládání této hry.

Konzole je textově založené rozhraní, v tomto případě je využito amerického standardního kódu pro výměnu informací, tedy ASCII, pro vykreslování informací cílovému uživateli.

### 4.1. Tvorba a vykreslování herní plochy

Při otevření konzolové aplikace je zavolána metoda, která má za úkol inicializovat herní plochu a přeměřovat hráče do fáze, kdy mohou rozmísťovat své lodě. Než je program převeden do této fáze, začne postupně vypisovat strukturu herní plochy, znak po znaku. První, co se objeví na obrazovce, je již zmíněný nadpis „*BATTLESHIP*,“ následovaný ohraničením herní plochy a jejími pomocnými souřadnicemi, které usnadňují orientaci hráče.

V průběhu výpisu jsou souřadnice reprezentovány klasickým značením, kde sloupce jsou označeny písmeny A-J a řádky číslicemi 0-9. Mřížka je tvořena vertikálními čárkami „/“ a tečkami, které symbolizují neobsazená pole. Kromě toho se na herní ploše objevuje i speciální znak reprezentující kurzor, který je viditelný pouze na hráčově poli. V konzolovém rozhraní jsou zobrazeny dvě takové herní pole – jedno patřící hráči a druhé AI, která v tomto případě představuje protihráče.

Ve stejném čase je inicializován dvojdimenzionální list, reprezentující pole hráče a umělé inteligence, který je dále užíván při interakcích a kalkulacích. Po této operaci je hra přeměřována do fáze rozmístění lodí a program se přeměřovává na volání metody určené k aktualizaci plochy v reálném čase.

Aktualizace plochy, či vykreslování v reálném čase, je velmi podobné jako tvorba samotná. Rozdílem je přítomnost dat, která byla vytvořena v předchozí metodě. Tyto data jsou využita ve smyčkách, jenž kontrolují, zda právě označené pole programem je loď, či ne. Pokud je nějaká část lodě nalezena, je označena příslušným písmenem a barvou.

Zmíněný proces je navržen tak, aby uživateli poskytoval jasnou a interaktivní prezentaci herního prostředí, které zahrnuje vše od identifikace hracích ploch až po vizuální nápovědy pro snadnější orientaci v průběhu hry. Příkladem jsou kontrasty a variace barev.

## 4.2. Detekce stisknutých kláves

Hned po tom, co je vytvořena herní plocha, program spustí nekonečnou smyčku asynchronně detekující stisknuté klávesy. Očekávané charaktery jsou reprezentovány v hexadecimálním soustavě, jenž referuje na charaktery ASCII.

Nejjednodušší ze všech detekcí jsou pohybové klávesy „WASD“ (0x57, 0x41, 0x53, 0x44), což je mechanickým standardem v herním průmyslu. Jakmile je jedna z těchto kláves stisknuta, poslední uložená souřadnice kurzoru je posunuta potřebným směrem.

Klávesa „ENTER“ je rozdělena na vícero operací. Charakter, na který kurzor právě směřuje, je přečten a uložen do paměti. Než se s tímto znakem pracuje, operace je vyfiltrována na právě nastavený typ lodě, jenž jde změnit pomocí klávesy „C,“ nebo je automaticky změněn, když hráč položí všechny části daného typu lodě. Pokud akce položení splní všechna kritéria pro rozmístění lodí, část je položena a pole aktualizováno.

„BACKSPACE“ je naprogramován podobně jako pokládání lodí. Jediným rozdílem je nepřítomnost kontroly typu lodí a kritérií rozmístění. Ta jsou nahrazena jedinou kontrolou, zda je označená část lodě mezi částmi jinými.

## 4.3. Rozmístění lodí

Po inicializaci herní plochy a zobrazení souřadnic se uživatel dostává do fáze rozmísťování svých lodí na hracím poli.

Před samotným rozmísťováním hráči vybírají typy lodí, které chtějí umístit na hrací pole. Typy lodí mohou zahrnovat ponorky, střední torpédoborce, větší křižníky a obří bitevní lodě. Každý typ má specifický počet polí, které zabírá na hrací ploše. V průběhu této fáze má hráč možnost měnit polohu svých lodí, pokud zjistí, že jejich původní rozmístění není optimální.



Hráč má možnost umísťovat lodě na herní plochu svisle nebo vodorovně. Tato volba je klíčová, protože ovlivňuje, jak dobře bude lodění skryto před protihráčem. Správná strategie spočívá v kombinaci různých orientací lodí, aby bylo pro soupeře obtížné odhadnout polohu lodí.

Při rozmísťování lodí je důležité zajistit, aby žádné dvě lodě nekolidovaly, nepřekrývaly se, nebyly umístěny diagonálně a nepřesahovaly plochu. To je zajištěno metodou fungující na bázi následující:

- 1) Kontrola, zda **je** loď stejného typu **okolo**.
- 2) Kontrola, zda loď stejného typu **není** směru diagonálního, **klesajíc**.
- 3) Kontrola, zda loď stejného typu **není** směru diagonálního, **stoupajíc**.
- 4) Kontrola, zda loď stejného typu **není v jiné lokaci** na hrací ploše.
- 5) Jestliže jedna z podmínek neplatí, program vyhodnotí, že hráč loď položit nemůže, tedy se nic nestane.

## 4.4. Střelba

Přechodem do fáze střelby je spuštěn cyklus, který očekává vstup od uživatele. Ten je očekáván ve formě souřadnic, jenž lze zadat postupně, nebo dohromady. Existuje také možnost tzv. „*stackování útoků*“, což umožňuje hráči plánovat a následně spustit sérii útoků najednou. Hráč musí však brát v úvahu, že protivník nebude ignorován a po každém útoku je na řadě právě on.

Pokaždé, co se hráč dostane na řadu, je požádán o souřadnice, na které plánuje vystřelit. Tyto souřadnice mají formát „[0-9][A-J]“, například „9A.“ Písmena jsou převedena na malá a konvertována do číslic. Následně je charakter v tomto prostoru uložen do paměti a porovnáván. Je-li objevena část lodě, je označena červeným „X“, opačně zůstane nezbarvená.

Příkladem funkce zásobníku může být „0a1b4d8e0a.“ Forma takového vstupu je postupně čtena programem, protože proměnné držící data souřadnic dokážou držet pouze jeden znak či číslo. Dokud jsou souřadnice validní, budou čteny až do konce.

## ZÁVĚR

Hlavním cílem této práce bylo vytvořit algoritmus, který by byl natolik kvalitní, že by bylo obtížné odlišit jeho rozhodovací procesy od těch reálného hráče, tedy člověka. Tento cíl byl dosažen díky detailnímu pochopení pochodu myšlení průměrného hráče této minihry, námořní bitvy.

V průběhu práce byla také provedena důkladná analýza pravidel Námořní bitvy, která umožnila plné porozumění dynamice hry. Následně byl vytvořen propracovaný algoritmus, schopný strategicky rozmisťovat lodě a provádět útoky s cílem maximalizovat šance na vítězství. Pro hráče bylo také vytvořeno interaktivní a uživatelsky přívětivé rozhraní pro rozmisťování lodí, či střelení.

Hlavní logika umělé inteligence však mohla být více propracovaná a nespolehat jen na nejběžnější pochody myšlení lidí. Kdyby byla využita funkce hustoty pravděpodobnosti, průměrný počet tahů by klesl o necelých 30 procent, tím by však umělá inteligence byla skoro nepřekonatelná, alespoň pro průměrné hráče.

Vyvinutý algoritmus je tedy dostačující a občas je v celku těžké ho i porazit. Nepoužívá jednoduché náhodné přístupy, ale logiku, která má nějaká pravidla. Důkladná analýza okolních oblastí se tedy v závěru vyplatila a práce dopadla podle očekávání autora práce.

## SEZNAM POUŽITÉ LITERATURY

- [1] MCCARTHY, John. WHAT IS ARTIFICIAL INTELLIGENCE? [online]. Dostupné z: <https://www-formal.stanford.edu/jmc/whatisai.pdf>
- [2] TURING, A. M. COMPUTING MACHINERY AND INTELLIGENCE [online]. Mind Journal, 1950 [cit. 2024-03-27]. Dostupné z: <https://www.ibm.com/topics/artificial-intelligence>. Článek. Oxford University Press.
- [3] RUSSELL, Stuart a Peter NORVIG. Approach, Artificial Intelligence: A Modern [online]. Dostupné z: <https://aima.cs.berkeley.edu>
- [4] IBM. What is artificial intelligence (AI)? [online]. Dostupné z: <https://www.ibm.com/topics/artificial-intelligence>
- [5] THE EDITORS OF ENCYCLOPAEDIA BRITANNICA. ASCII | Definition, History, Trivia, & Facts [online]. Dostupné z: <https://www.britannica.com/topic/ASCII>
- [6] PROGRAMMING IS FUN. Console Application ASCII Design [online]. Dostupné z: <https://programmingisfun.com/command-line-ascii-design/>
- [7] BROWN, Seth. How to Play the Battleship Board Game [online]. Dostupné z: <https://www.thesprucecrafts.com/the-basic-rules-of-battleship-411069>
- [8] BERRY, Nick. Battleship [online]. Dostupné z: <http://datagenetics.com/blog/december32011/index.html>
- [9] VAN DER LAKEN, Over Paul. Beating Battleships with Algorithms and AI [online]. Dostupné z: <https://paulvanderlaken.com/2019/01/21/beating-battleships-with-algorithms-and-ai/>

## SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

ASCII American Standard Code for Information Interchange

AI Umělá Inteligence / Artificial Intelligence

## SEZNAM OBRÁZKŮ

Obrázek č. 1 – Uživatelské rozhraní pro konzolovou hru.....	11
Obrázek č. 2 – Příklad fáze rozložení lodí .....	12
Obrázek č. 3 – Příklad odstřelovací fáze hry .....	13
Obrázek č. 4 – Příklad konce hry .....	14
Obrázek č. 5 – Příklad logiky odstřelení lodě umělou inteligencí.....	18
Obrázek č. 6 – Princip šachovnice [8] .....	19
Obrázek č. 7 – Výpočet pravděpodobnosti, že dlaždice obsahuje loď, na základě všech možných rozložení hrací plochy [8] .....	20
Obrázek č. 8 – Příklad distribuce se sedmi ranami na hracím poli [8] .....	21
Obrázek č. 9 – Matice pravděpodobnosti jako tepelná mapa pro každé pole po každém tahu ve hře [9] .....	22