

<i>Autor</i>	<i>Przedmiot</i>	<i>Prowadzący</i>	<i>Temat</i>	<i>Data</i>
<i>Maciej Molik (molikmaciek@student.agh.edu.pl)</i>	Języki programowania	<i>Prof. Bogusław Cyganek</i>	<i>Gra komputerowa</i>	<i>25.06.2023</i>

# ***Wojna Morska***

# **WSTĘP**

*Dokument dotyczy opracowania gry komputerowej pt. "Wojna Morska".*

*Celem tego projektu jest stworzenie programu do gry w statki.*

## **FUNKcjONALNOŚĆ**

*Wojna Morska będzie obejmować interaktywny interfejs, w którym gracze będą mogli:*

- *Rozmieszczać swoje statki na planszy*
- *Wybierać pola do ataku i próbować trafić statki przeciwnika*
- *Otrzymywać informacje zwrotne o trafieniach i pudłach*
- *Kontynuować grę aż do zniszczenia wszystkich statków przeciwnika*

## **ANALIZA PROBLEMU**

*Gra w statki ma na celu zapewnienie rozrywki i rywalizacji dla dwóch graczy. Wymaga rozmieszczenia statków na planszy i odgadywania położenia statków przeciwnika.*

*Żeby rozwiązać powyższy problem zaproponowałem użycie klas :*

**Gra** - służąca do stworzenia całego środowiska gry, to w tej klasie tworzona jest interakcja z użytkownikiem i wywołane są 4 obiekty klasy Plansza jako mapy obu graczy i mapy wykonywanych działań.

**Plansza** – w tej klasie tworzona jest cała obsługa mapy gry, tworzone są tu też obiekty typu wyliczeniowego Pole które są fragmentami planszy do gry. Klasa służy też do zapisywania postępów gry.

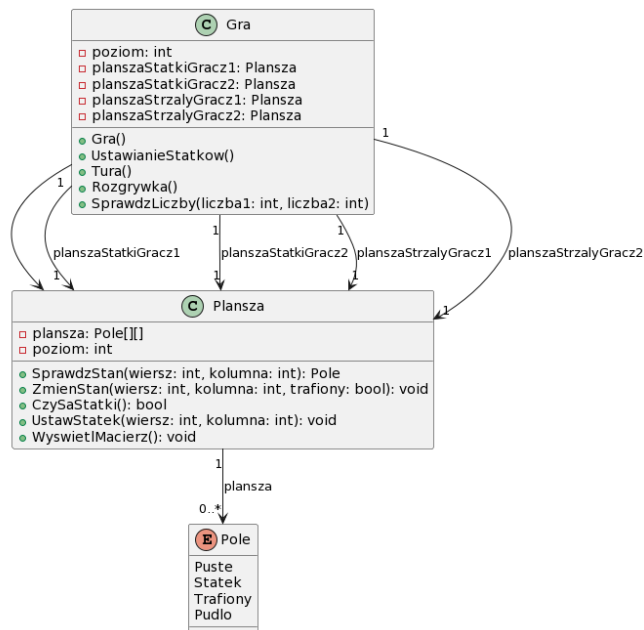
*I trybu wyliczeniowego :*

**Pole** – określa stan pojedynczego pola planszy .

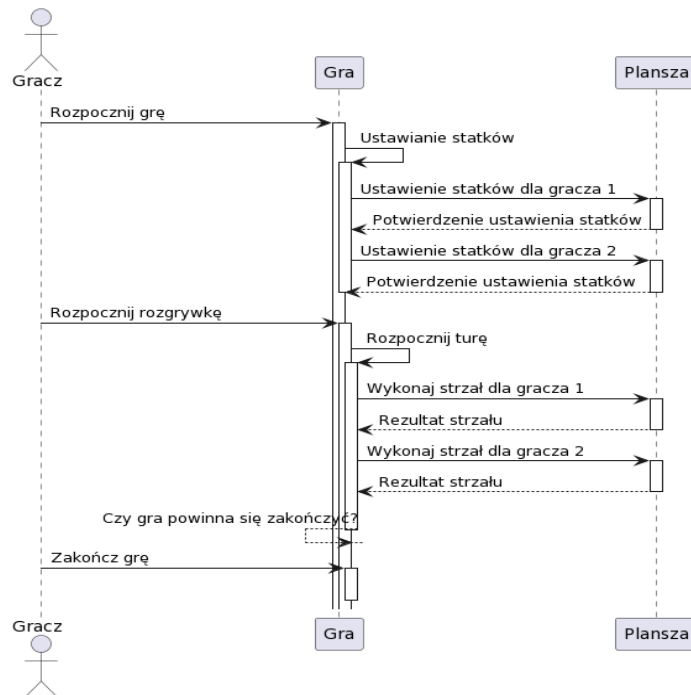
*Moja implementacja powyższego problemu jest pokazana w punkcie 4 .jako diagram klas i diagram uml.*

# PROJEKT TECHNICZNY

**Diagram klas** – przedstawia zależności między klasami w projekcie .



**Diagram UML** – przedstawia przepływ informacji w projekcie.



## ***OPIS REALIZACJI***

*Projekt zrealizowałem za pomocą poniższych narzędzi:*

- *Komputer stacjonarny*
- *Visual studio*
- *Języka C++*
- *Plantuml*
- *Biblioteki (vector, iostream)*

*Projekt zrealizowałem w poniższych krokach*

1. *Rozplanowanie działania gry.*
2. *Projektowanie Diagramu klas.*
3. *Projektowanie sposobu przepływu informacji w programie zgodnego z logiką gry.*
4. *Implementacja powyższych punktów w języku C++.*

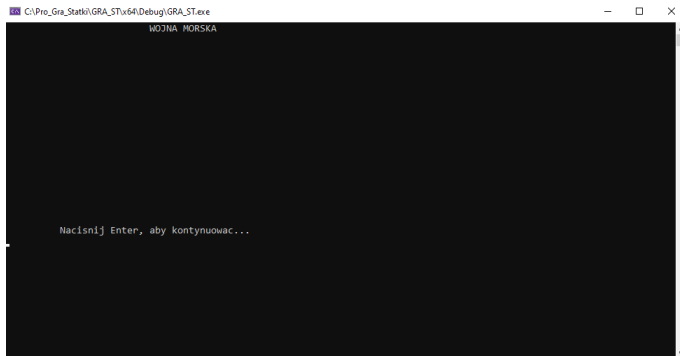
## ***OPIS WYKONANYCH TESTÓW***

<i>Kod usterki</i>	<i>Opis</i>	<i>Stan</i>
<i>brak</i>	<i>Podczas wpisywania współrzędnych ustawianego statku lub wykonywanego strzału gdy wpisany został znak lub liczba z poza planszy dalsze używanie programu było niemożliwe</i>	<i>Naprawione -została wprowadzona funkcja która sprawdza wpisane przez użytkownika znaki</i>


*Testy zostały wykonane pod kątem sprawdzenia awaryjności programu gdy zostaną podane niewłaściwe i nieprzewidziane przez autora dane przez użytkownika*

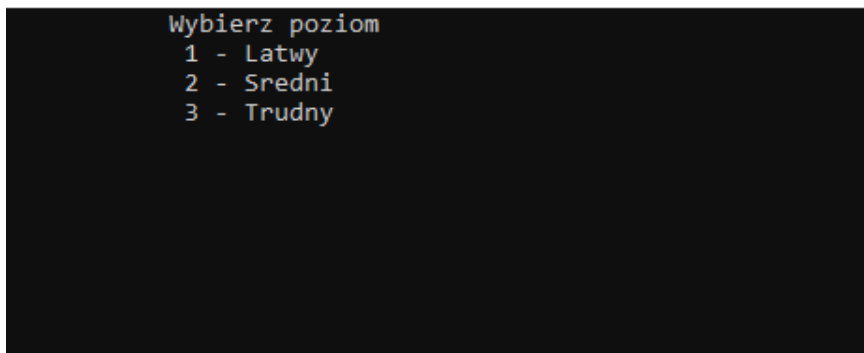
# PODREČZNIK UŻYTKOWNIKA

1. Aby rozpocząć grę należy wcisnąć Enter.

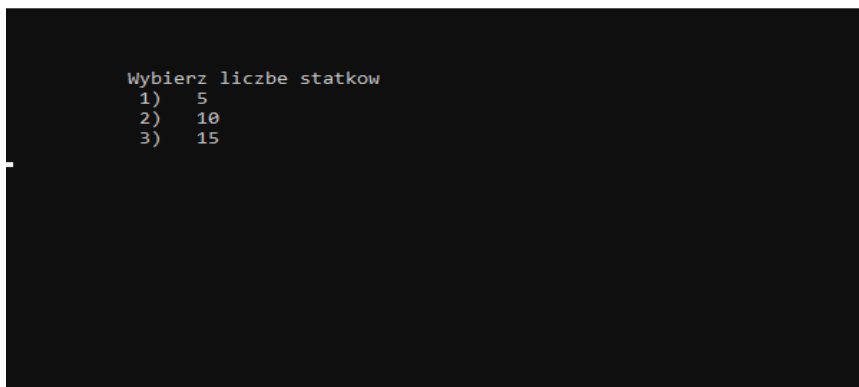


2. Żeby dostosować poziom trudności należy wybrać jedną z opcji która odzwierciedla wielkość mapy a później należy wybrać ilość okrętów.

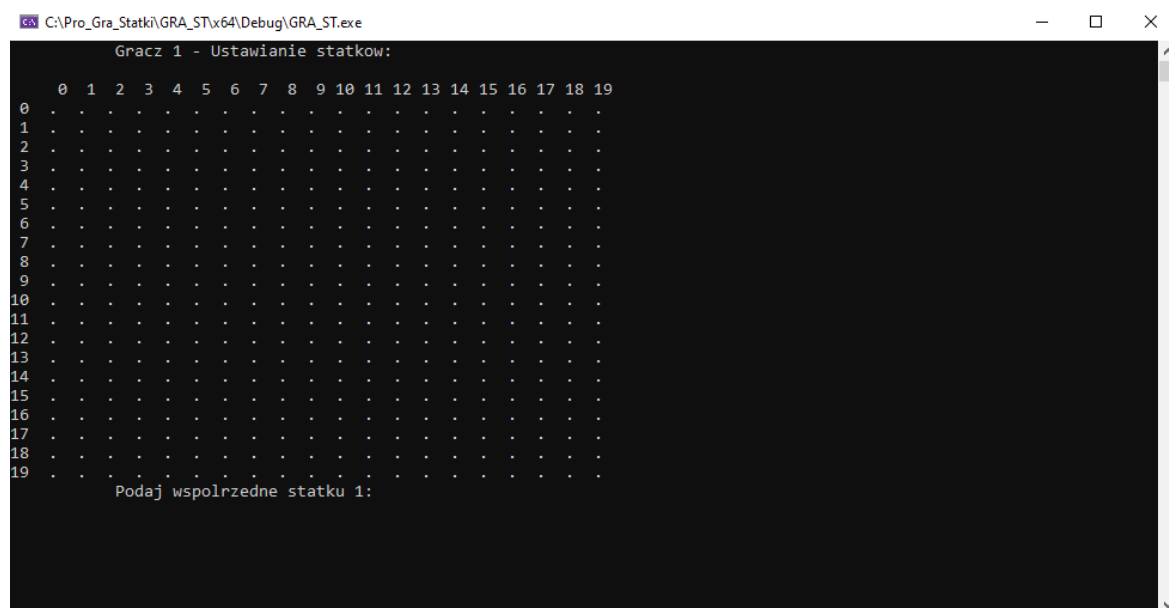
 C:\Pro\_Gra\_Statki\GRA\_ST\x64\Debug\GRA\_ST.exe



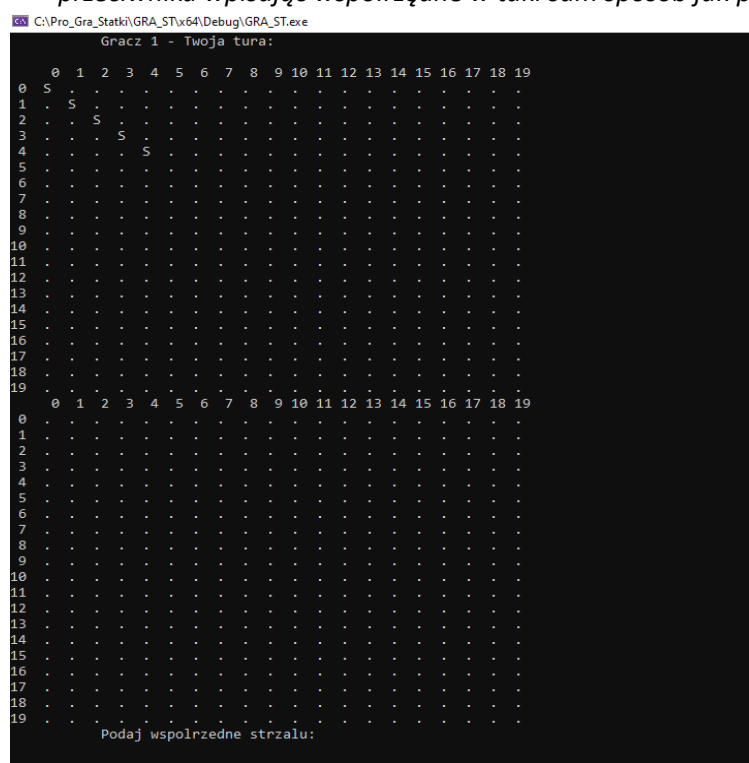
 C:\Pro\_Gra\_Statki\GRA\_ST\x64\Debug\GRA\_ST.exe



3. Po dostosowaniu poziomu trudności należy ustawić swoje okręty, polega to na wybraniu pierwszej współrzędnej, naciśnięcie enter i podaniu drugiej współrzędnej. Czynność należy powtarzać aż do ustawienia wszystkich statków (analogicznie dla gracza 2).



4. Jak okręty są już ustawione na pozycjach zaczyna się gra. W tej fazie gry należy strzelać w okręty przeciwnika wpisując współrzędne w taki sam sposób jak podczas ustawiania statków



## BIBLIOGRAFIA

<https://learn.microsoft.com/pl-pl/>  
<https://plantuml.com/>