

Politechnika Śląska  
Wydział Matematyki Stosowanej  
Kierunek Informatyka

Gliwice, 20.01.2020

Programowanie I  
**projekt zaliczeniowy**

**"Gra w Statki"**

**Dominik Szczepanik gr. lab. 4**

**1. Opis projektu.**

*Projektem jest gra w statki, pomiędzy graczem i komputerem. Zwycięża strona która zatopi wszystkie okręty przeciwnika.*

## **2. Wymagania**

- Wyświetlanie dwóch planszy w czasie rzeczywistym. Statki przeciwnika będą ukryte do momentu trafienia. Gracz będzie mógł obserwować stan swojej planszy i pozostałych mu statków.*
- Prosta sztuczna inteligencja komputera.*
- Możliwość uzupełniania swojej planszy ręcznie jak i automatycznie.*
- Plansza komputera generowana losowo.*
- Zliczanie trafień/zatopień.*
- Przyjazna oprawa graficzna.*
- Instrukcja gry.*
- Odgłosy pola walki oraz inne dźwięki.*

## **3. Przebieg realizacji**

*Program zawarty jest w funkcji `gra()`, która jest funkcją główną, zawierającą wszystkie najważniejsze etapy gry.*

*Gdy zaczynamy rozgrywkę ukazuje nam się ekran startowy za który odpowiada funkcja `EkranStartowy()`. Następnie gdy zdecydujemy się zagrać, mamy przed sobą ekran wyboru pomiędzy rozmieszczeniem naszych statków automatycznie bądź własnoręcznie; zarządza tym funkcja `EkranStartowy2()`. Określając sposób rozmieszczenia funkcja zwraca odpowiednią wartość dla następnego kroku czyli funkcji rozmieszczeń; te funkcje to `rozmiescStatkiRecznie()` oraz `rozmiescStatkiAutomatycznie()`.*

*Ostatnim etapem po rozmieszczeniu swoich statków, jest automatyczne rozmieszczenie statków komputera i rozpoczęcie właściwej rozgrywki z nim.*

*Rozgrywka jest umieszczona w pętli while() z warunkiem zakończenia gry, która kończy się w momencie gdy liczba statków przeciwnika lub nasza spadnie do zera. Pętla posiada wszystkie niezbędne elementy:*

- wyświetlanie mapy statków i odświeżanie jej*
- wpisywanie współrzędnych do strzału*
- zapisywanie miejsc trafień lub chybień*
- zabezpieczenia przed błędami podczas wpisywania*
- określenie wyniku gry*

*Po tym jak funkcja gra() określi wygranego ukazuje nam się ekran z napisem naszej wygranej lub przegranej, możemy rozpocząć po tym kolejną rozgrywkę lub ją zakończyć.*

*Rozpoczynając kolejną rozgrywkę plansze gracza i komputera, plansze zaznaczające miejsca trafień oraz punktacja są resetowane aby rozgrywka była możliwa.*

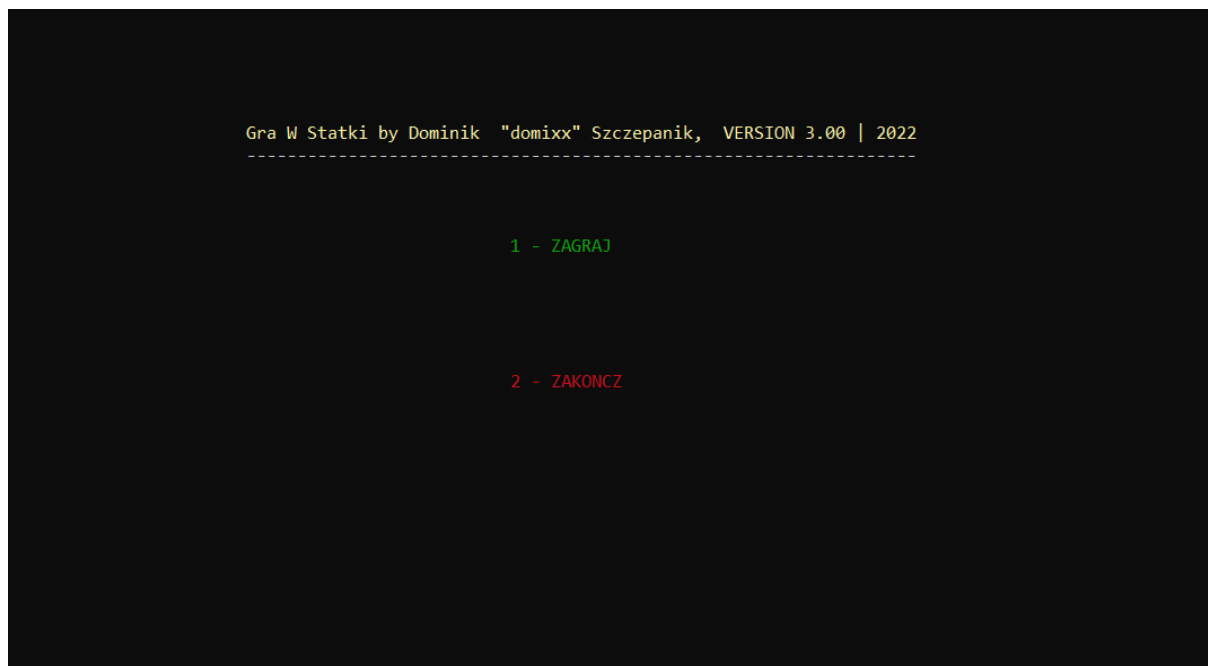
*Program zawiera również proste funkcje które pomagają szybciej i czytelniej uporządkować kod w funkcjach bardziej rozbudowanych.*

*Cały program znajduje się w jednym pliku projektowym. Znajdują się w nim również efekty dźwiękowe użyte do urozmaicenia rozgrywki.*

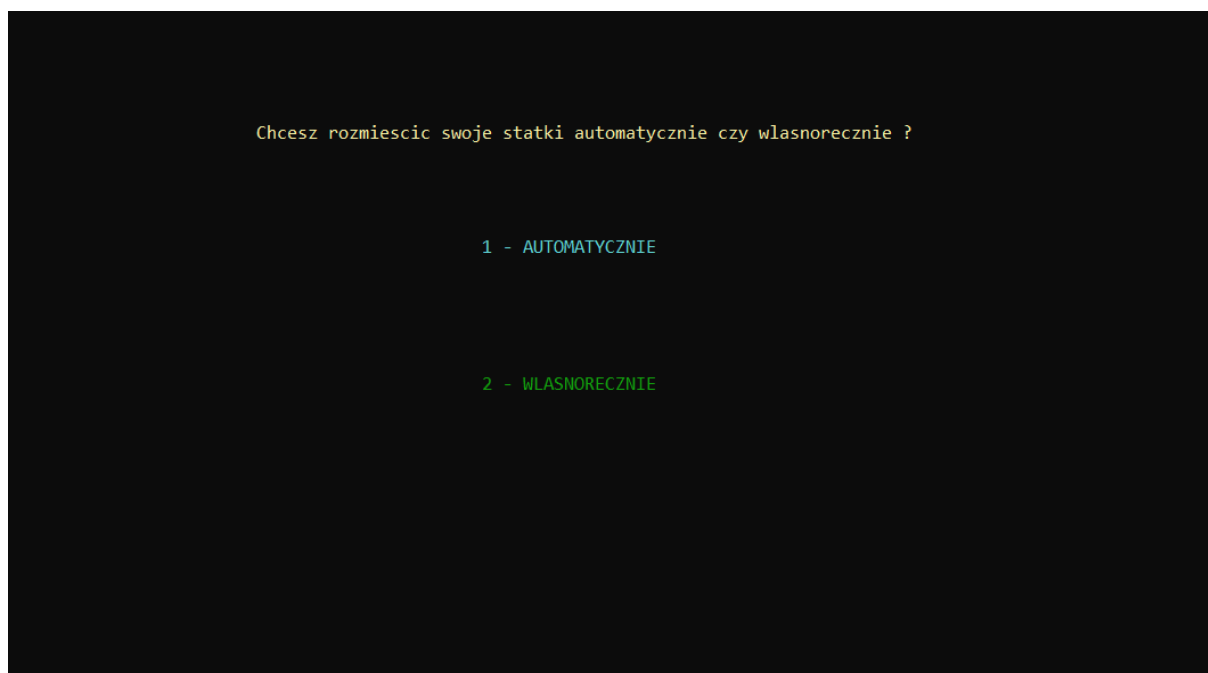
*Wykorzystane biblioteki :*

- random (do losowania miejsc)*
- ctime (do opóźnienia podczas efektów)*
- conio.h (użycie funkcji getch() do odczytu znaku podczas wciśnięcia)*
- iomanip (do formatowania tekstu)*
- Windows.h (odtwarzanie dźwięków oraz użycie kolorów)*
- string (zamiana zmiennej typu int na string)*

## 4. Instrukcja użytkownika



Ekran startowy - zagraj/zakończ grę.



Wybieramy czy chcemy rozmieścić nasze statki automatycznie czy własnoręcznie.

	A	B	C	D	E	F	G	H	I	J
[1]	▲	■	■	■	▲	⌘	▲	■	▲	⌘
[2]	▲	▲	▲	▲	▲	⌘	▲	■	▲	⌘
[3]	▲	▲	▲	▲	▲	▲	▲	■	▲	⌘
[4]	■	▲	■	▲	▲	■	▲	■	▲	⌘
[5]	▲	▲	■	▲	▲	■	▲	▲	▲	⌘
[6]	▲	▲	▲	▲	▲	▲	▲	⌘	⌘	⌘
[7]	■	▲	▲	▲	■	▲	▲	▲	▲	⌘
[8]	▲	▲	■	▲	▲	▲	▲	■	▲	⌘
[9]	⌘	▲	■	▲	■	▲	▲	■	▲	⌘
[10]	⌘	▲	▲	▲	▲	▲	▲	■	▲	⌘

Statek - ■ Miejsce Zabronione - ▲  
Czy podany układ ci odpowiada ? Jesli tak naciśnij T jesli nie to N

Rozmieszczenie statków automatycznie - gdy chcemy zmienić układ naciskamy N, jeśli nam odpowiada naciskamy T.

	A	B	C	D	E	F	G	H	I	J
[1]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
[2]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
[3]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
[4]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
[5]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
[6]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
[7]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
[8]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
[9]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
[10]	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘

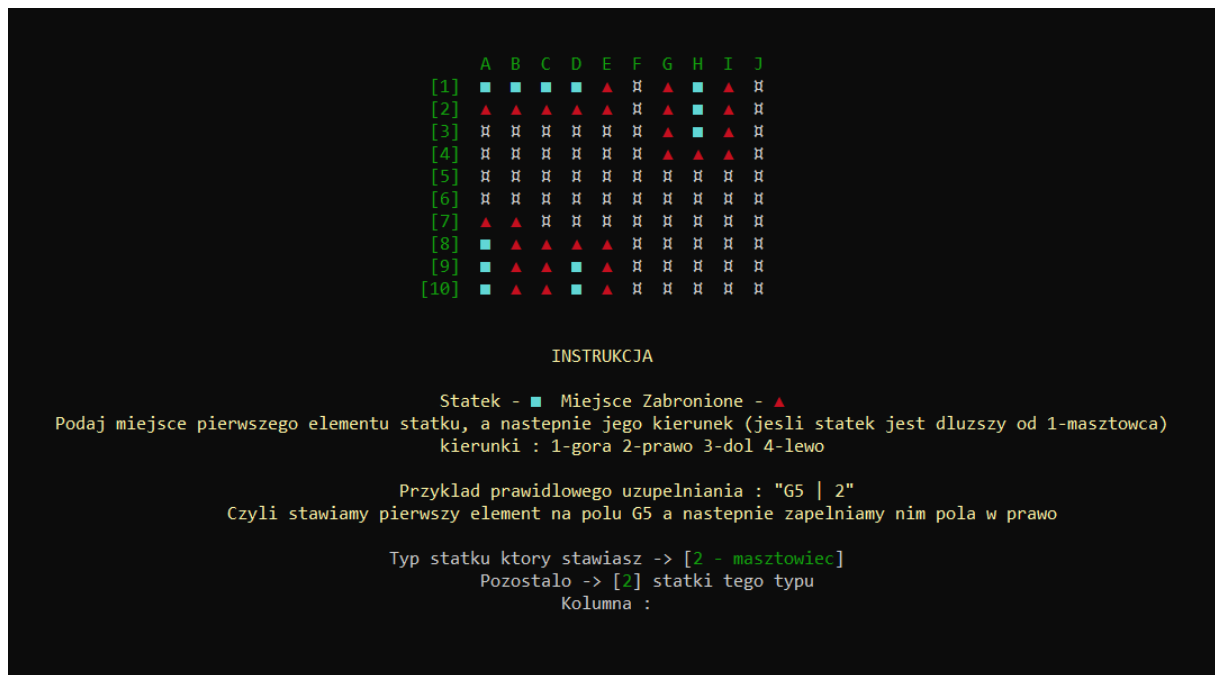
#### INSTRUKCJA

Statek - ■ Miejsce Zabronione - ▲  
Podaj miejsce pierwszego elementu statku, a następnie jego kierunek (jesli statek jest dluzszy od 1-masztowca)  
kierunki : 1-gora 2-prawo 3-dol 4-lewo

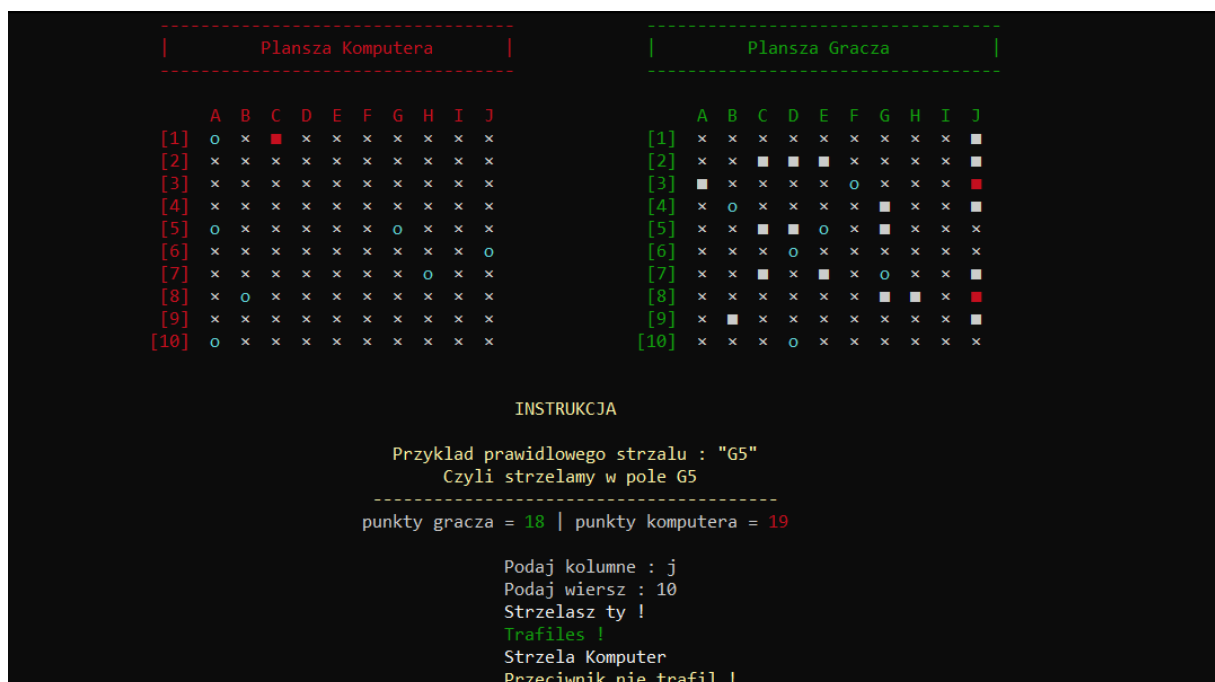
Przykład prawidłowego uzupełniania : "G5 | 2"  
Czyli stawiamy pierwszy element na polu G5 a następnie zapełniamy nim pola w prawo

Typ statku który stawiasz -> [4 - masztowiec]  
Pozostal 1 statek tego typu  
Kolumna :

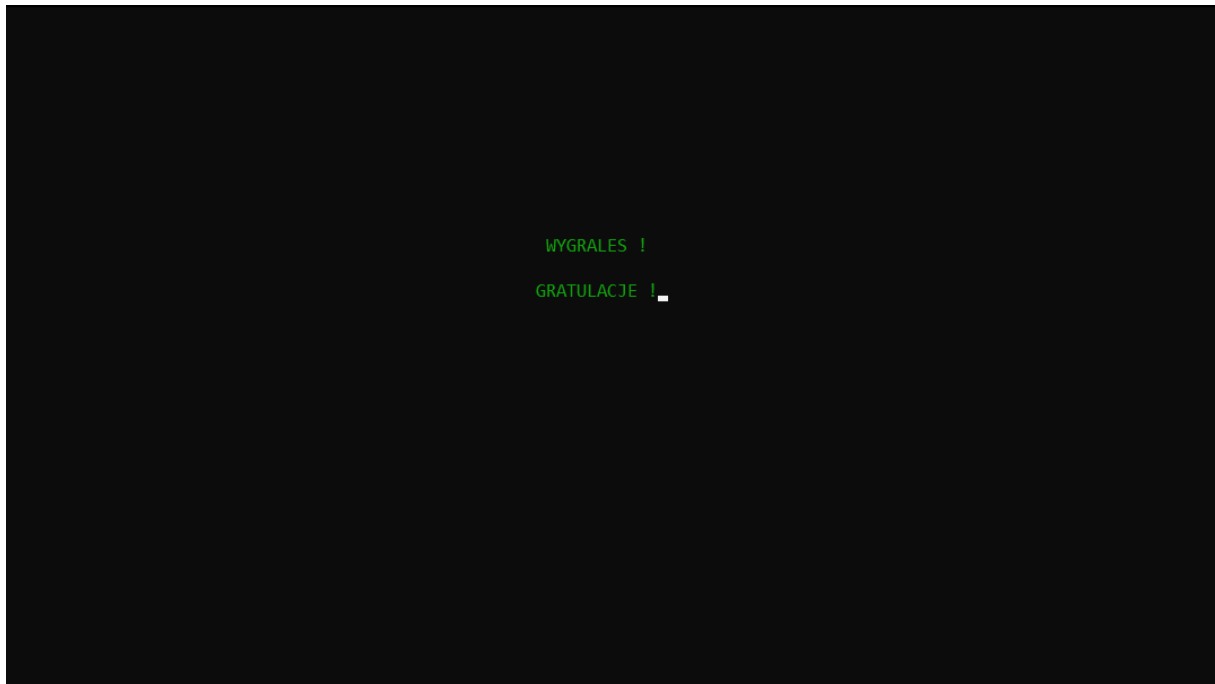
Zapełnianie planszy własnoręcznie przez gracza - Widzimy długość statku który zapełniamy oraz ile statków tego typu pozostało do zapełnienia .



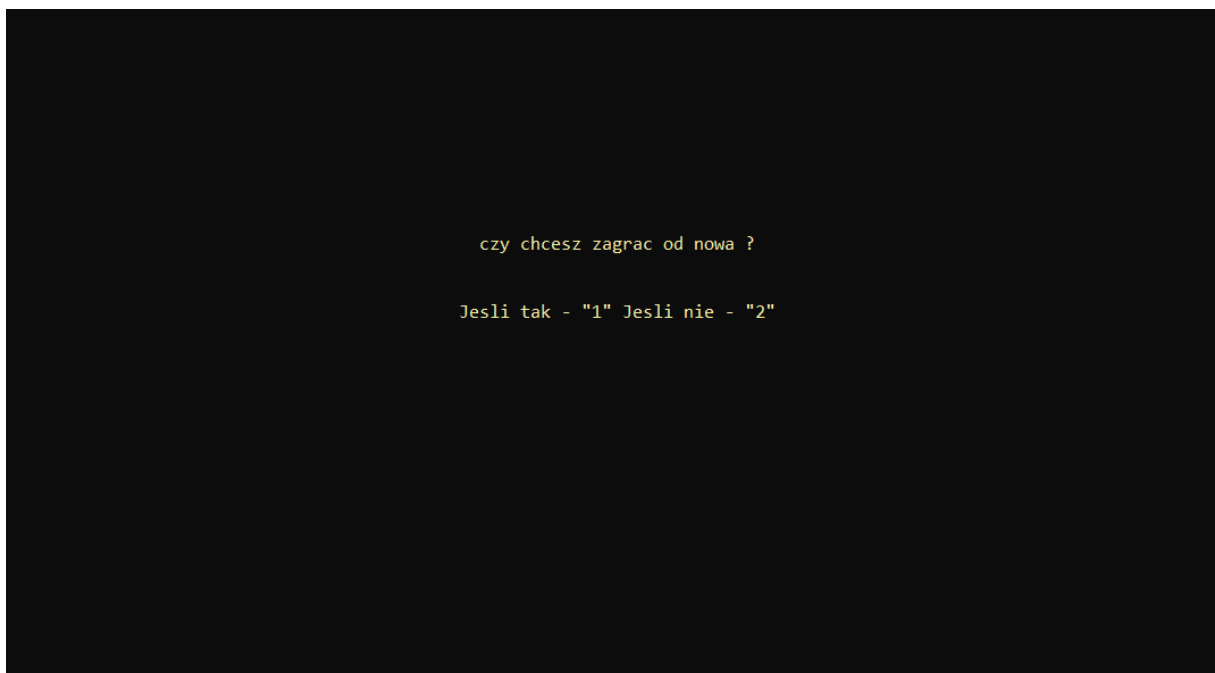
Zapełnianie planszy w trakcie.



Właściwa rozgrywka – wybieramy pole w które chcemy strzelić a następnie oddajemy strzał; po strzale dostajemy informacje zwrotną czy trafiliśmy; następnie komputer wykonuje swój ruch. Gra się kończy jeśli punkty któreś ze stron spadną do zera.



Ekran końcowy(w tym przypadku wygrana)



Po ekranie końcowym mamy możliwość zagrania jeszcze raz.

## 5. Podsumowanie i wnioski.

*Zostały zrealizowane prawie wszystkie założenia; liczba zatopionych statków nie pojawiła się w programie, zamiast tego pojawił się licznik punktów trafień.*

*Najwięcej czasu poświęciłem algorytmie sortowania automatycznego aby był jak najbardziej efektywny(szybki).*

*Mnóstwo czasu poświęciłem również na estetykę, dźwięk oraz interakcję użytkownika z grą.*

*Uczyłem się na bieżąco wielu rzeczy i zacząłem komentować każdy ważniejszy element kodu, aby później wiedzieć jakie jest jego przeznaczenie.*

*Gdy napotykałem problem korzystałem z debugger'a aby go zlokalizować. Przeszukiwałem również fora internetowe aby uzyskać jak najlepsze rozwiązania na moje problemy.*

*Kierunek w którym projekt mógłby być rozwijany to poprawienie sztucznej inteligencji oraz statystyki po potyczce z komputerem.*

*Projekt był wymagający aczkolwiek poświęciłem mu dużo czasu i nauczyłem się sporo nowych rzeczy.*