

Autor: Jakub Kajzer

Opiekun projektu: dr. inż. Marek Niewiński

Data oddania pracy do oceny: 14.01.2018 r.

## ZADANIE 8 PRM.B 2017Z

Napisać program umożliwiający grę w statki z komputerem.

Założenia:

- Rozmiar pola rozgrywania gry: definiowana przez użytkownika
- Ilość i rodzaj statków definiowana przez użytkownika
- Sposób ich rozmieszczania: dowolny
- Symulator gracza realizowany przez program powinien mieć wbudowaną „pewną logikę” tzn np. automatycznie znajdował pola na których nie mogą znajdować się statki przeciwnika

## FUNKCJE W PROGRAMIE

Nazwa pliku	Typ funkcji	Nazwa funkcji	Opis funkcji
engine.c	void	AI_strzel()	Funkcja symulująca oddawanie strzału przez przeciwnika.
	void	losuj_statek()	Funkcja ustawiają statki na planszy przeciwnika
	int	main()	Funkcja główna programu
	int	policz_statki_gracza()	Funkcja zwracająca niezatopione statki gracza
	int	policz_statki_przeciwnika()	Funkcja zwracająca niezatopione statki przeciwnika
	void	strzel()	Funkcja pozwalająca na oddanie graczowi strzału na wskazane pole
	int	ustaw_statek()	Funkcja umożliwiająca ustawianie statków na planszy
	void	wyswietl_dwa()	Funkcja wyświetlająca planszę gracza i planszę trafień w przeciwnika
	void	wyswietl_jedno()	Funkcja wyświetlająca jedną planszę
main.c	void	logo()	Funkcja wyświetlająca logo w menu
	int	main()	Funkcja główna programu
	void	menu()	Funkcja wyświetlająca menu główne
	void	newgame()	Funkcja uruchamiająca engine z argumentem
	void	options()	Funkcja wyświetlająca menu opcji

## ZMIENNE W PROGRAMIE

Nazwa pliku	Nazwa funkcji	Typ zmiennej	Nazwa zmiennej	Opis zmiennej
engine.c	void wyswietl_jedno()	int	i	Licznik stosowany w pętli
		int	i	Licznik stosowany w pętli
	void wyswietl_dwa()	int	j	Licznik stosowany w pętli
		int	j	Licznik stosowany w pętli
	int ustaw_statek()	int	czyblad	Flaga przechowująca czy statek poprawnie ustawiono
		int	dlugosc	Zmienna przechowująca rozmiar statku
		char	dummy	Zmienna używana do czyszczenia bufora wejścia ze znaków
		int	err	Flaga przechowująca czy należy podać współrzędne jeszcze raz w wyniku nieprawidłowych danych
		int	i	Licznik stosowany w pętli
		int	j	Licznik stosowany w pętli
		int	k	Zmienna przechowująca kierunek statku
		int	wczytane	Zmienna przechowująca ilość poprawnie wczytanych danych
		int	x	Zmienna przechowująca współrzędną x pola
		int	y	Zmienna przechowująca współrzędną y pola
	void losuj_statek()	int	czyblad	Flaga przechowująca czy statek poprawnie ustawiono
		int	err	Flaga przechowująca czy należy podać współrzędne jeszcze raz w wyniku nieprawidłowych danych
		int	i	Licznik stosowany w pętli
		int	j	Licznik stosowany w pętli
		int	k	Zmienna przechowująca kierunek statku
		int	x	Zmienna przechowująca współrzędną x pola

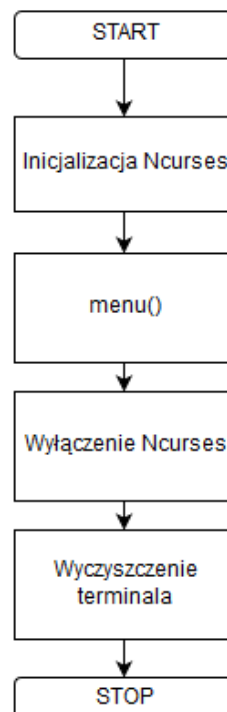
		int	y	Zmienna przechowująca współrzędną y pola
	void strzel()	char	dummy	Zmienna używana do czyszczenia bufora wejścia ze znaków
		int	wczytane	Zmienna przechowująca ilość poprawnie wczytanych danych
		int	x	Zmienna przechowująca współrzędną x pola
		int	y	Zmienna przechowująca współrzędną y pola
	int policz_statki_gracza()	int	i	Licznik stosowany w pętli
		int	j	Licznik stosowany w pętli
		int	suma	Zmienna przechowująca sumę pól zajmowanych przez statki
	int policz_statki_przeciwnika()	int	i	Licznik stosowany w pętli
		int	j	Licznik stosowany w pętli
		int	strzelone	Zmienna przechowująca sumę trafiony pól ze statkami
		int	suma	Zmienna przechowująca sumę pól zajmowanych przez statki
	void AI_strzel()	static int	czytrafione	Zmienna odpowiedzialna za tryb strzelania komputera
		static int	offset	Zmienna przechowująca odległość od pierwszego trafienia w statek
		static int	ost_x	Zmienna przechowująca współrzędną x trafionego statku gracza
		static int	ost_y	Zmienna przechowująca współrzędną y trafionego statku gracza
		static int	pionowepodejrzenia	Zmienna przechowująca rozpoznanie kierunku ustawienia statku gracza
		int	x	Zmienna przechowująca współrzędną x pola
		int	y	Zmienna przechowująca współrzędną y pola
	int main()	char	dummy	Zmienna używana do czyszczenia bufora wejścia ze znaków
		int **	gracz	Tablica dwuwymiarowa rozmieszczenia statków gracza

		int	i	Licznik stosowany w pętli
		int	j	Licznik stosowany w pętli
		int	liczbastatkow	Zmienna przechowująca ilość statków na planszy
		int **	ostrzal	Tablica dwuwymiarowa strzałów i trafień gracza w przeciwnika
		int **	przeciwnik	Tablica dwuwymiarowa rozmieszczenia statków przeciwnika
		int	rozmiar	Zmienna przechowująca rozmiar plansz
		int[]	statki	Tablica jednowymiarowa przechowująca ilość statków o danym rozmiarze
		int	wczytane	Zmienna przechowująca ilość poprawnie wczytanych danych
main.c	void options()	int	kolumny	Zmienna przechowująca ilość kolumn terminala
		int	ktory	Zmienna przechowująca aktualną pozycję
		const short int	max_wybor	Zmienna przechowująca maksymalną pozycję wyboru
		const short int	min_wybor	Zmienna przechowująca minimalną pozycję wyboru
		const char	txt1[]	Stała tablica znaków przechowująca tekst
		const char	txt2[]	Stała tablica znaków przechowująca tekst
		const char	txt3[]	Stała tablica znaków przechowująca tekst
		const char	txt4[]	Stała tablica znaków przechowująca tekst
		const char	txt5[]	Stała tablica znaków przechowująca tekst
		int	wiersze	Zmienna przechowująca ilość wierszy terminala
		int	znak	Zmienna pomocnicza do obsługi klawiatury
	void logo()	int	kolumny	Zmienna przechowująca ilość kolumn terminala
		int	wiersze	Zmienna przechowująca ilość wierszy terminala
	void menu()	int	kolumny	Zmienna przechowująca ilość kolumn terminala

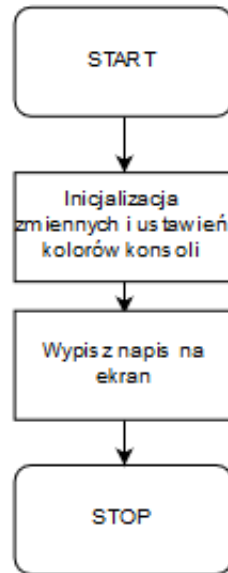
		int	ktory	Zmienna przechowująca aktualną pozycję
		const short int	max_wybor	Zmienna przechowująca maksymalną pozycję wyboru
		const short int	min_wybor	Zmienna przechowująca minimalną pozycję wyboru
		const char	txt1[]	Stała tablica znaków przechowująca tekst
		const char	txt2[]	Stała tablica znaków przechowująca tekst
		const char	txt3[]	Stała tablica znaków przechowująca tekst
		int	wiersze	Zmienna przechowująca ilość wierszy terminala
		int	znak	Zmienna pomocnicza do obsługi klawiatury
	ZMIENNA GLOBALNA	int	rozmiar	Zmienna przechowująca rozmiar planszy

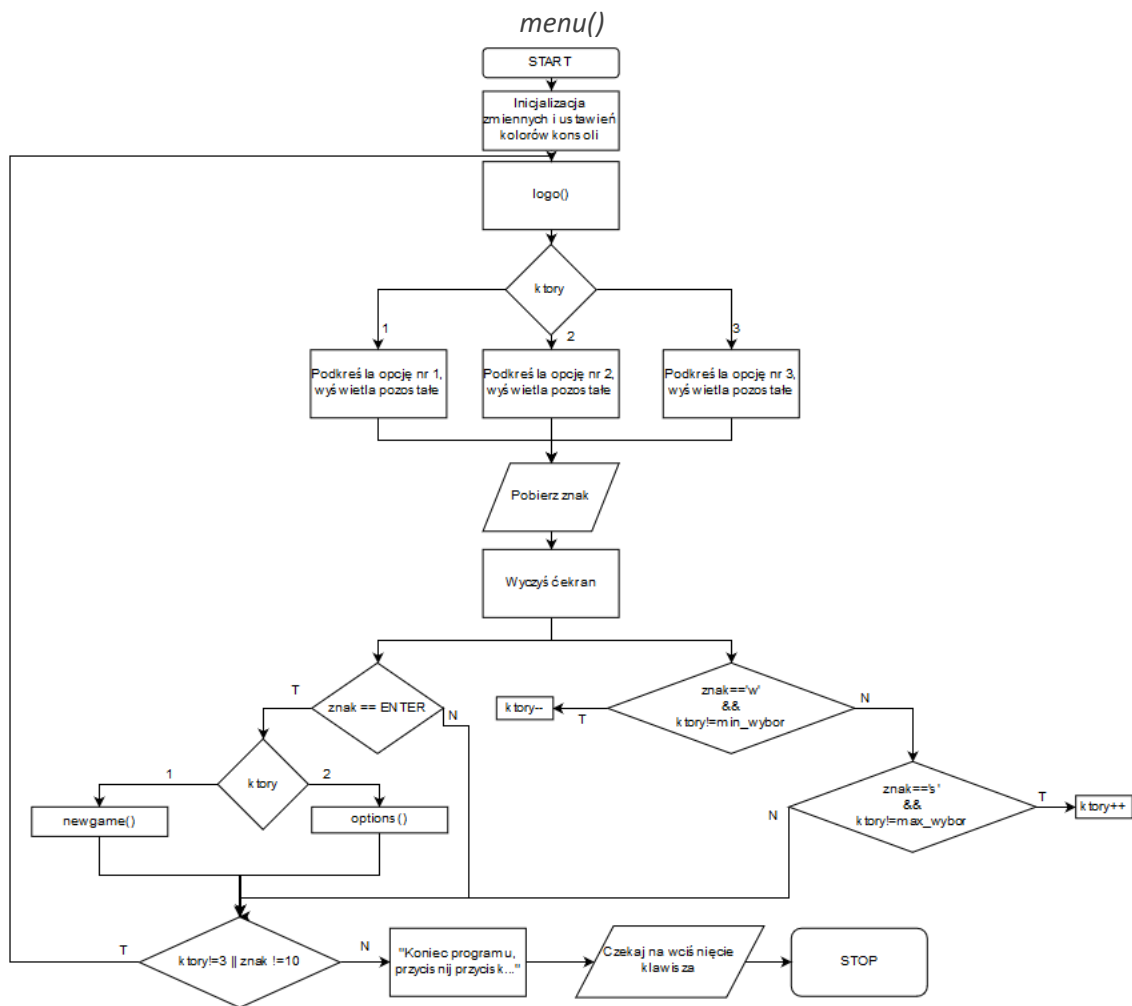
## FUNKCJE – Plik main.c

*main()*

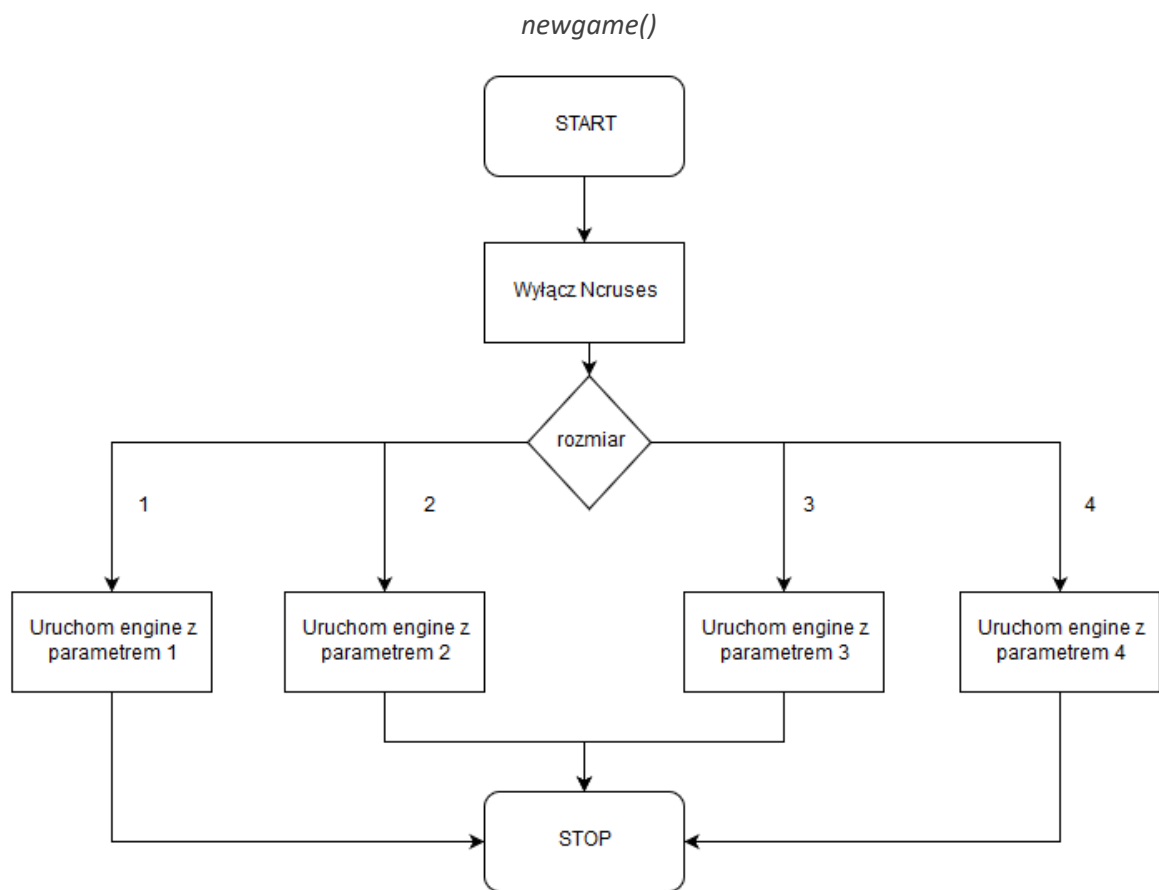


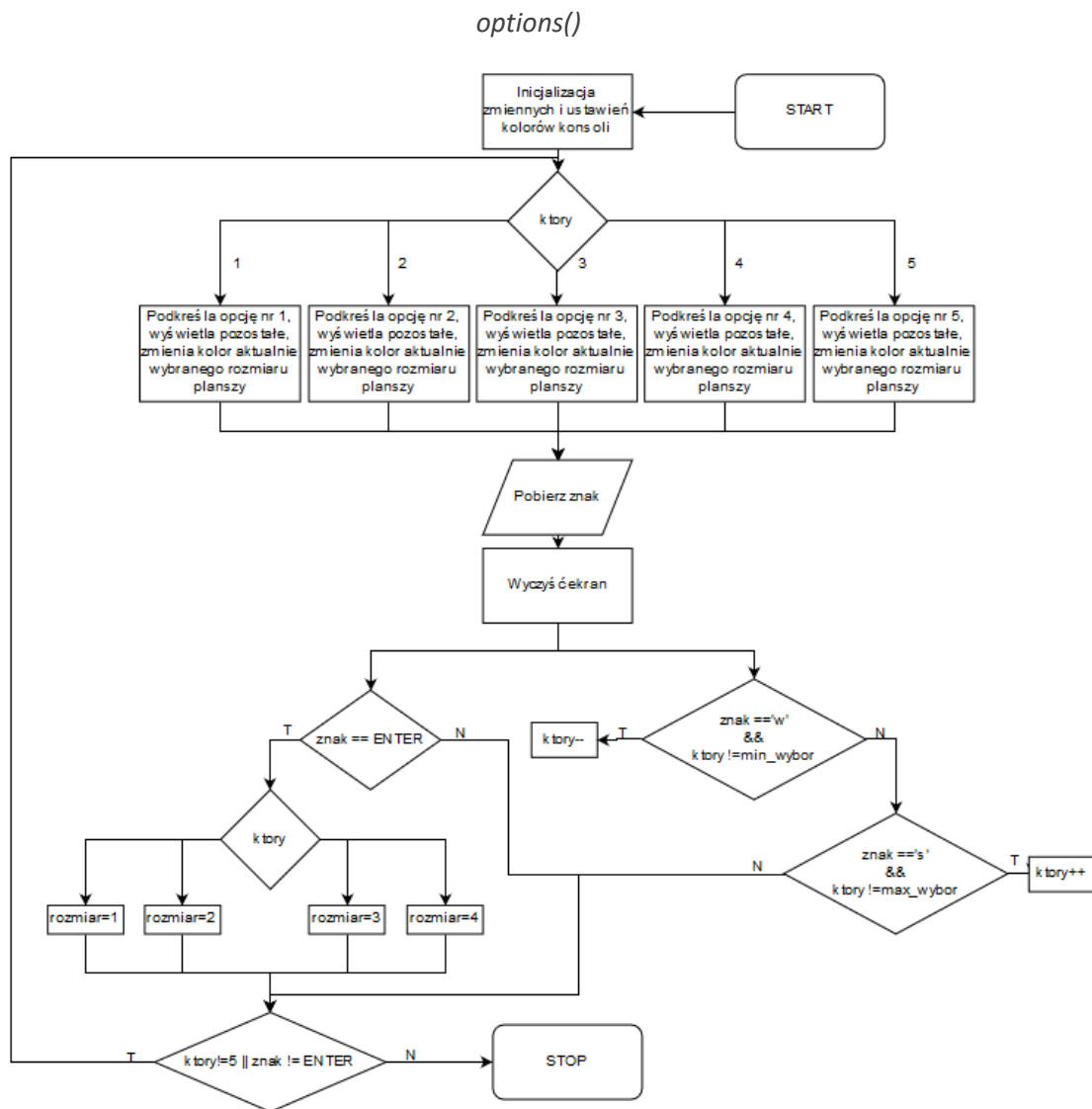
*logo()*





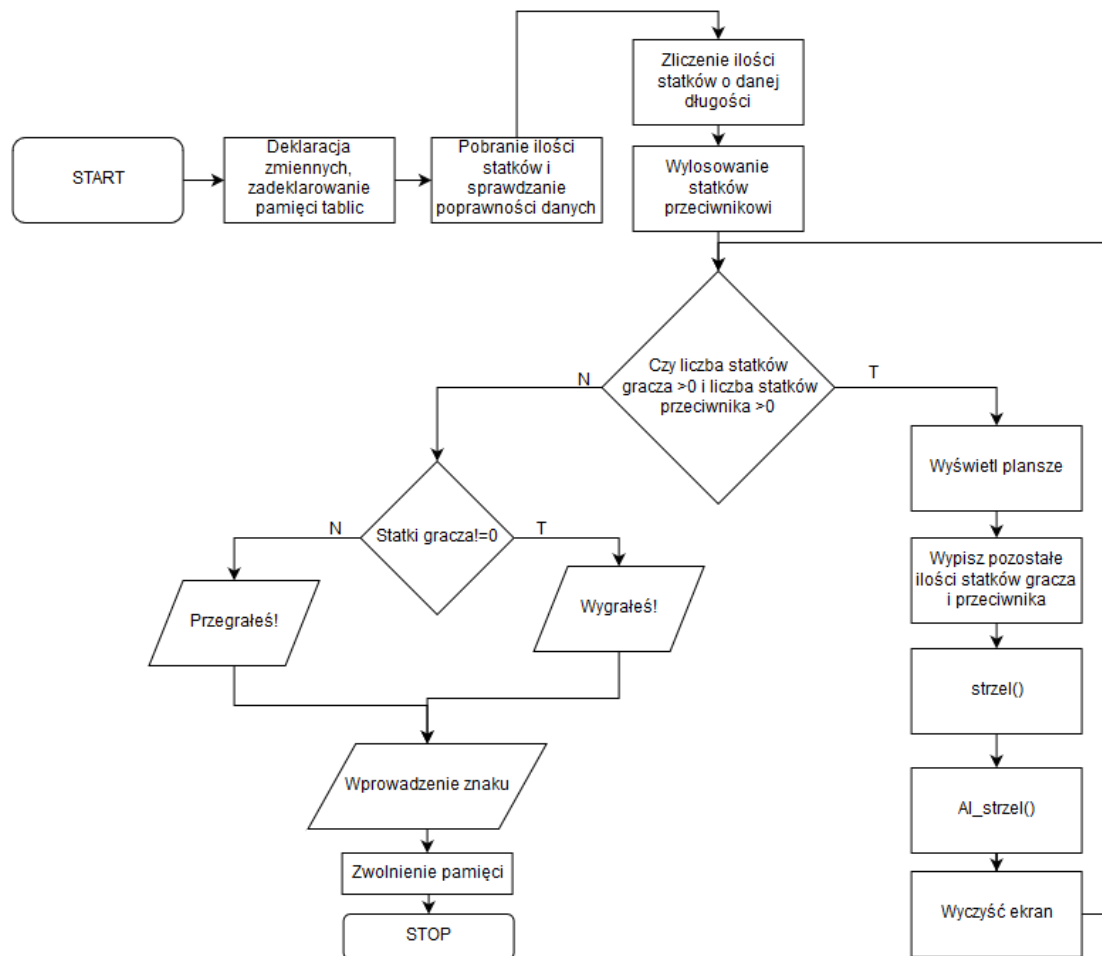




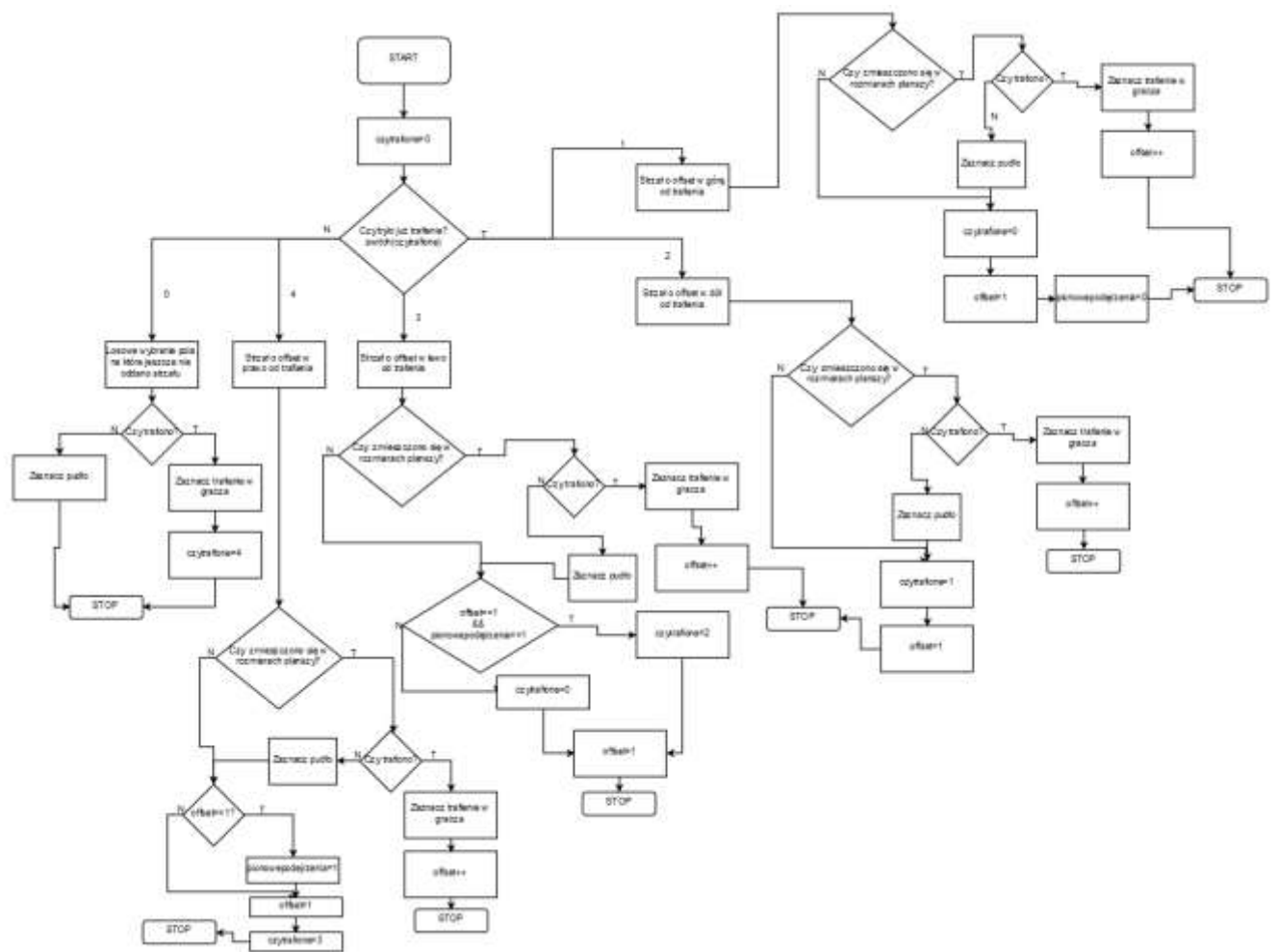


# FUNKCJE – Plik engine.c

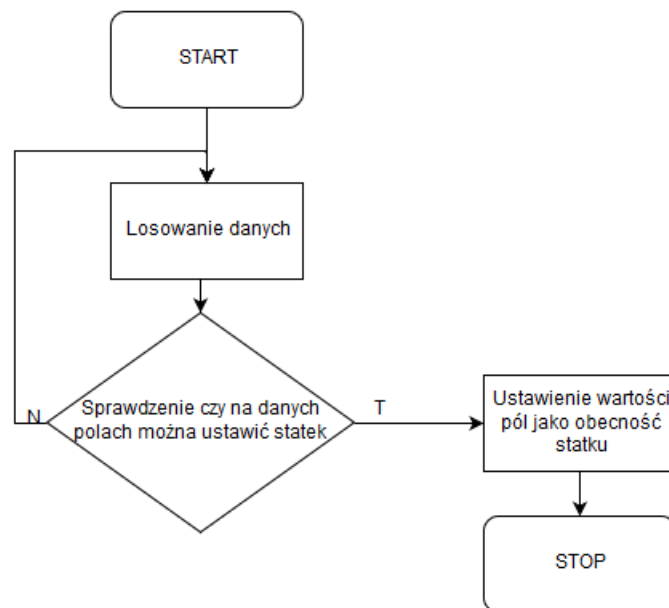
*main()*



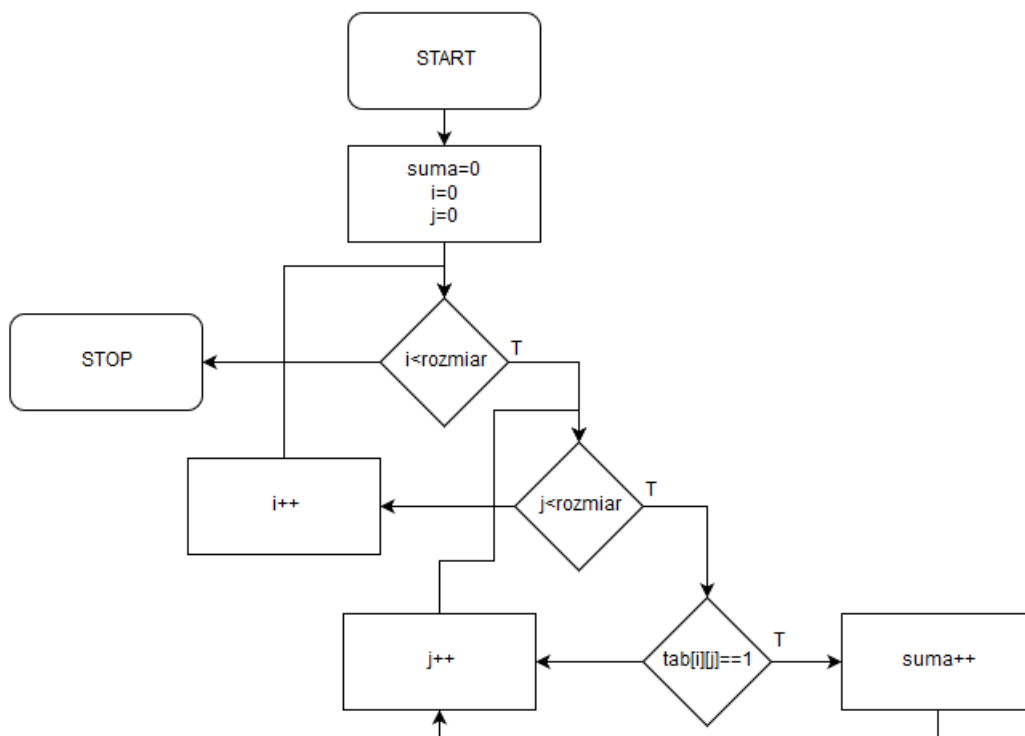
*Al\_strzel()*



*losuj\_statek()*

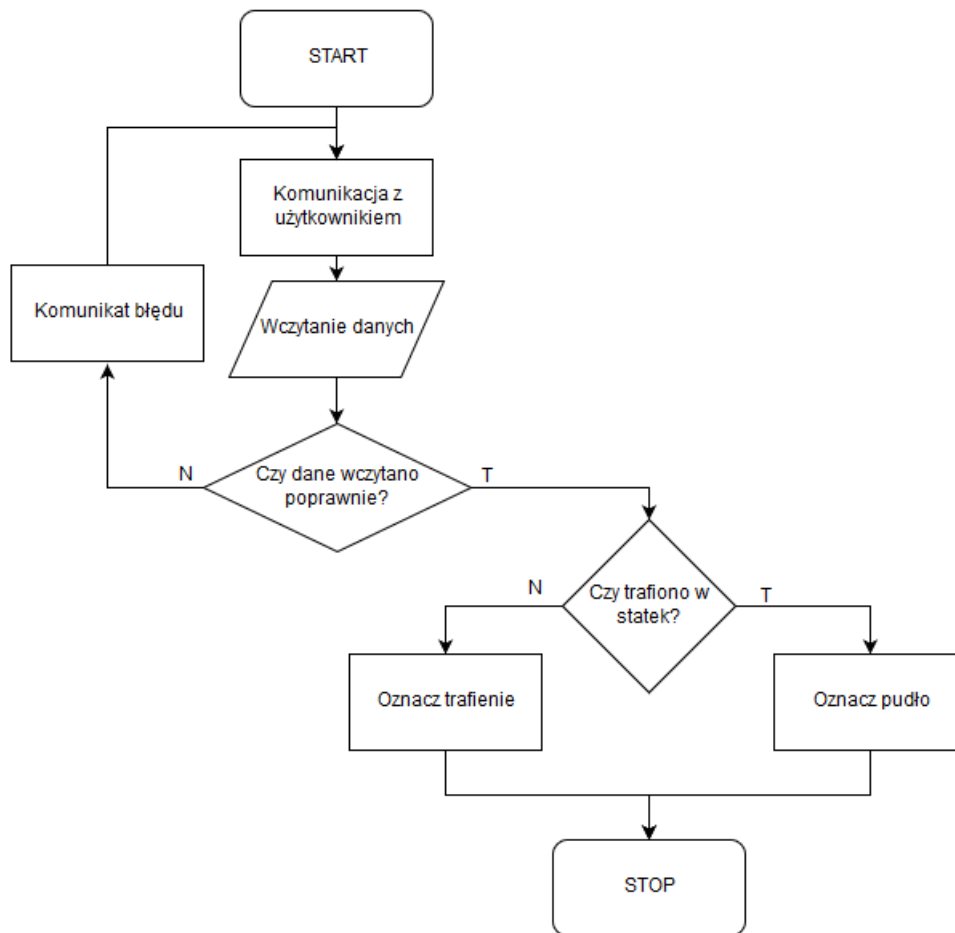


*policz\_statki\_gracza()*

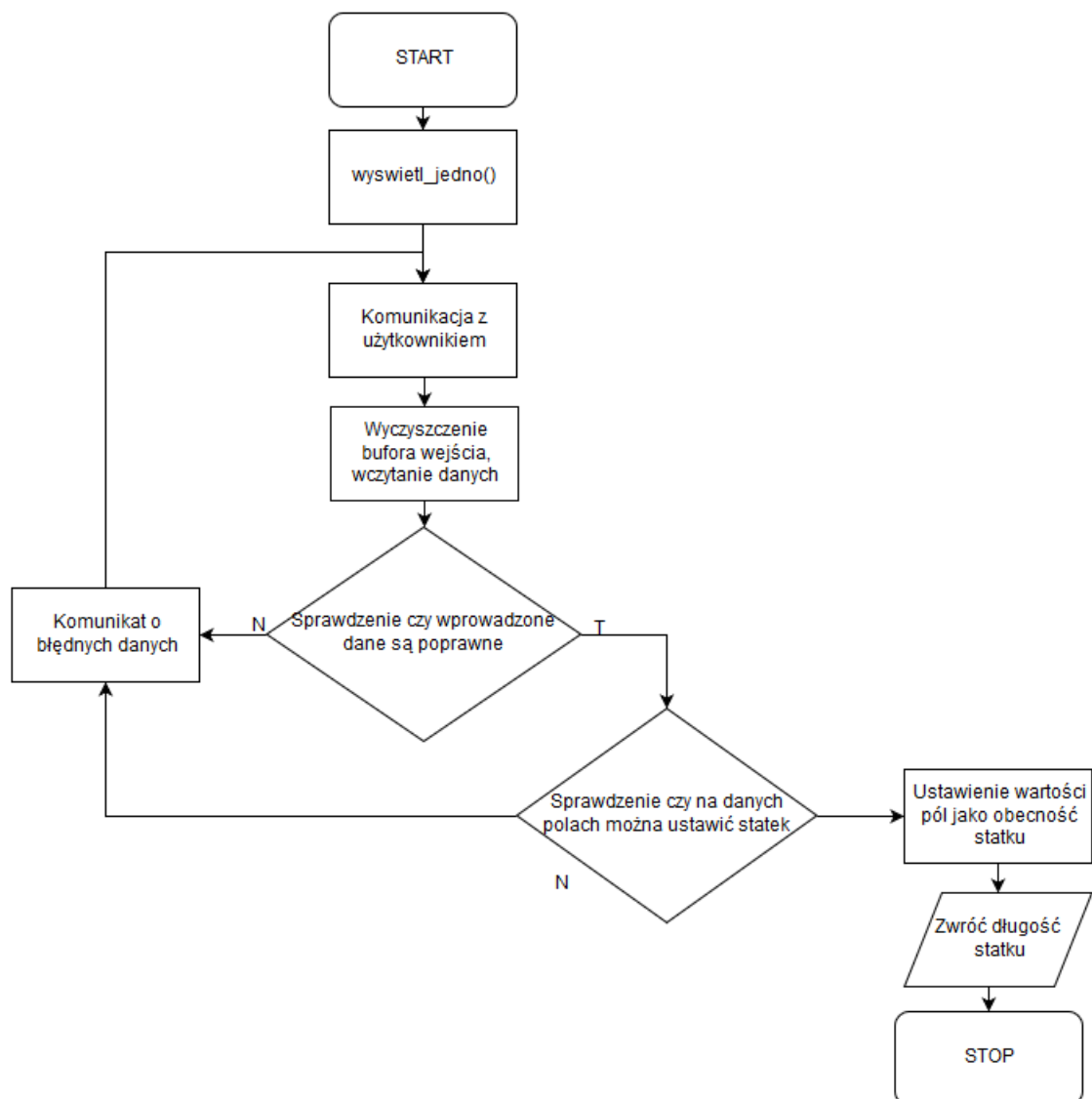


Uwaga. Funkcja *policz\_statki\_przeciwnika* jest analogiczna do *policz\_statki\_gracza()* i dlatego nie została tu uwzględniona, różnicą jest zliczenie z dwóch tablic i odjęcie ich od siebie.

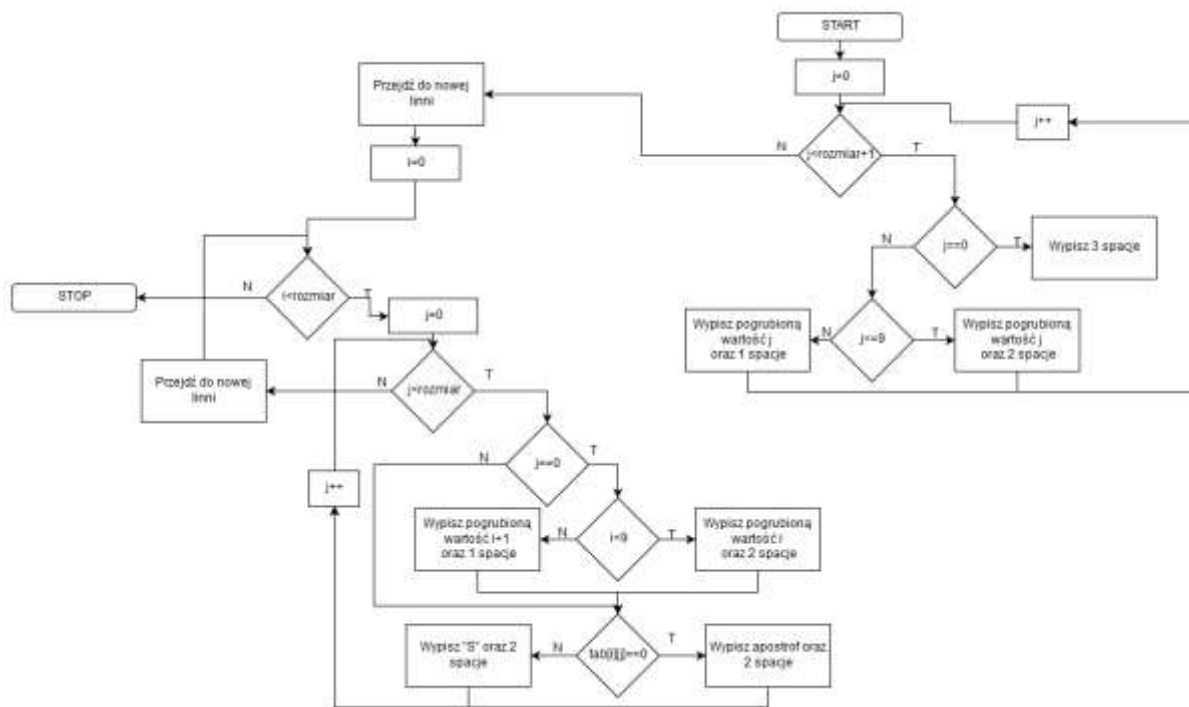
*strzel()*



*ustaw\_statek()*



wyswietl\_jedno()



Uwaga. Funkcja wyswietl\_dwa jest analogiczna do wyswietl\_jedno() i dlatego nie została tu uwzględniona, różnicą jest wyświetlenie drugiej tablicy obok pierwszej.



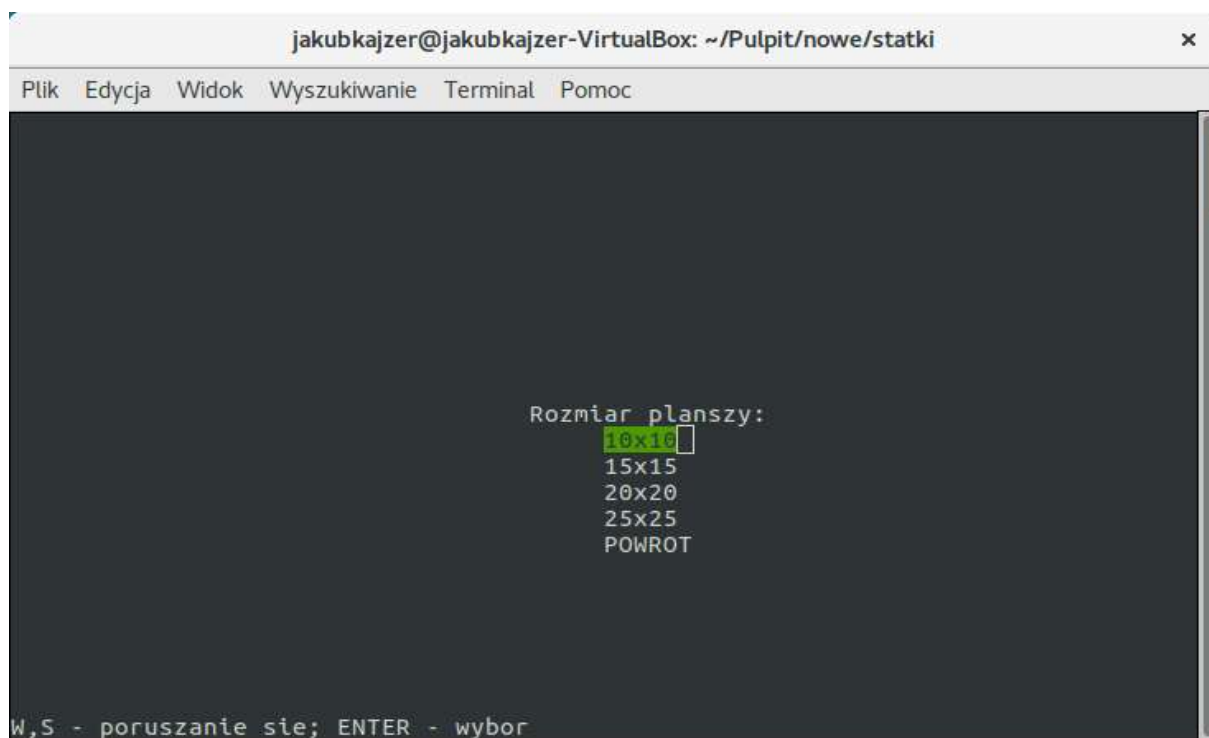
## DZIAŁANIE PROGRAMU

Program uruchamiany jest poprzez skompilowany plik main.c, plik engine.c powinien zostać skompilowany pod nazwą „engine” w celu poprawnego działania programu.

Menu wygląda tak jak pokazano na zrzucie ekranu.



Wybór rozmiaru w menu



Wygląd graficzny ustawiania statków.

```
jakubkajzer@jakubkajzer-VirtualBox: ~/Pulpit/nowe/statki
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

 1  2  3  4  5  6  7  8  9 10
1  |  |  |  |  |  |  |  |  |  |
2  |  |  |  |  |  |  |  |  |  |
3  |  |  S  |  |  |  |  |  |  |
4  |  |  |  S  |  |  |  |  |  |
5  |  |  |  S  |  |  |  |  |  |
6  |  |  |  S  |  |  |  |  |  |
7  |  |  |  S  |  |  |  |  |  |
8  |  |  |  S  |  |  |  |  |  |
9  |  |  |  |  |  |  S  S  |  |
10 |  |  |  |  |  |  |  |  |  |

Podaj wsporzedne pola poczatku statku, jego kierunek(1 w prawo, 0 w dol) i dlugo
sc(1-5)
W postaci:(x y k d):
```

Wygląd właściwej rozgrywki, kontrola niewłaściwego wprowadzania danych również działa.

```
jakubkajzer@jakubkajzer-VirtualBox: ~/Pulpit/nowe/statki
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

 1  2  3  4  5  6  7  8  9 10
1  |  |  |  |  |  |  S  S  S
2  |  |  |  |  |  |  |  |  |
3  |  |  S  |  |  |  |  |  |
4  |  |  |  S  |  |  |  |  |
5  |  |  |  S  |  |  |  |  |
6  |  |  |  S  |  |  P  |  |
7  |  |  P  X  P  |  |  |  |
8  |  |  |  X  |  |  |  |  |
9  |  |  |  P  |  |  S  S  |
10 |  |  |  |  |  |  |  |

 1  2  3  4  5  6  7  8  9 10
1  |  |  |  |  |  |  |  |  |
2  |  |  |  |  |  |  |  |  |
3  |  |  |  X  |  |  |  |  |
4  |  |  |  X  |  |  |  |  |
5  |  |  |  X  |  P  |  |  |
6  |  |  P  |  |  |  |  |  |
7  |  |  |  |  |  |  |  |  |
8  |  |  |  |  |  |  |  |  |
9  |  |  |  |  |  |  |  |  |
10 |  |  |  |  |  |  |  |  |

Pola statkow gracza: 9
Pola statkow przeciwnika: 9
Podaj wspolrzedne pola ostrzalu:24
4
Podales bledne wspolrzedne pola!
Podaj wspolrzedne pola ostrzalu:f
Podales bledne wspolrzedne pola!
Podaj wspolrzedne pola ostrzalu:
```

Tu zaprezentowano działanie algorytmu znajdowania statków gracza.

```
jakubkajzer@jakubkajzer-VirtualBox: ~/Pulpit/nowe/statki
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

  1  2  3  4  5  6  7  8  9  10
1  '  '  '  '  '  '  '  S  S  S
2  '  '  '  '  '  '  '  '  '  '
3  '  '  S  P  '  '  '  '  '  '
4  '  '  '  X  '  '  '  '  '  '
5  '  '  '  X  '  '  '  '  '  '
6  '  '  '  X  '  '  P  '  '  '
7  '  '  P  X  P  '  '  '  '  '
8  '  '  '  X  '  '  '  '  '  '
9  '  '  '  P  '  '  '  S  S  '
10 '  '  '  '  '  '  '  '  '  '

Pola statkow gracza: 6
Pola statkow przeciwnika: 8
Podaj wspolrzedne pola ostrzalu:
```