

# Projekt z języków symbolicznych

Dokumentacja

HUBERT WĄTORCZYK GL.08

## Temat projektu

Tematem projektu jest gra Okręty (ang. Battleship). Jest to gra strategiczno-planszowa dla dwóch osób. Gra jest znana w wielu krajach pod nazwą: gra w Statki lub Salvo. W polskiej literaturze można się też spotkać z nazwą: wojna morska lub bitwa morska.

## Funkcjonalność

- Okno z dwoma planszami 10x10 pól (np. siatki przycisków) oraz przyciskiem rozpoczęcia gry przyciskiem reset. Dodatkowo został umieszczony przycisk instrukcja, wyświetlający wskazówki jak należy grać.
- Na początku gracz rozmieszcza okręty (1x czteromasztowiec, 2x trójmasztowiec, 3x dwumasztowiec, 4x jednomasztowiec). Okręty są rozmieszczane przez wybór dwóch pól między którymi ma stanąć okręt. W przypadku jednomasztowca należy dwa razy wybrać to samo pole.
- Po rozmieszczeniu okrętów przez gracza i wciśnięciu przycisku „Bitwa” przeciwnik komputerowy losowo rozmieszcza swoje okręty.
- Okręty nie mogą się dotykać ani bokami, ani rogami.
- Po rozmieszczeniu okrętów przez obu graczy jeden z nich wykonuje pierwszy ruch (losowo gracz lub komputer).
- Wybór celu przez gracza następuje przez kliknięcie pola, w razie trafienia przycisk staje się czerwony, w przeciwnym razie niebieski (nie można strzelić dwa razy w to samo pole).
- Komputer strzela w losowe, nie wybrane wcześniej pole. Po trafieniu próba znalezienia orientacji statku i zestrzelenie go do końca.
- Gra kończy się, gdy któryś gracz straci ostatni okręt, wyświetlane jest okno z informacją o zwycięzcy.

## Moduł Battleships

### Klasa Game

Główna klasa odpowiadająca za okno gry

### Konstruktor

Tworzy główne okno programu, ramki, plansze, przyciski, etykiety, dwie instancje klasy Competitor z modułu objects, wyświetla napis powitalny

### Funkcje:

#### create\_buttons\_board

Tworzy planszę przycisków (10x10)

:param frame: ramka na której ma zostać umieszczona siatka

:param h: wysokość przycisku

:param w: szerokość przycisku

:param fun: funkcja wywoływana przez przycisk

:return: lista przycisków tkinter

#### [start\\_game](#)

Rozpoczyna grę. Wywoływana przez przycisk Start

#### [reset\\_game](#)

Resetuje grę. Wywoływana przez przycisk Reset

#### [get\\_player\\_ships\\_to\\_place](#)

Zwraca statki do rozstawienia dla gracza

#### [display\\_message](#)

Wyświetla komunikat w górnej części okna

:param message: treść komunikatu

#### [get\\_displayed\\_message](#)

Zwraca aktualnie wyświetlony komunikat

#### [get\\_computer\\_ships\\_coordinates](#)

Zwraca współrzędne staków komputera

#### [get\\_computer\\_board](#)

Zwraca plansze przycisków komputera

#### [place\\_ship](#)

Rozmieszcza statek użytkownika

:param coordinate: współrzędne x oraz y przycisku

#### [computer\\_shoot](#)

Wykonuje strzał komputera

#### [player\\_shoot](#)

Wykonuje strzał gracza

:param coordinate: współrzędne strzału

#### [end\\_game](#)

Kończy grę

:param name: nazwa wygranego

## **[Moduł objects](#)**

### **[Klasa Ship](#)**

Klasa reprezentująca pojedynczy statek

#### [Konstruktor](#)

Uzupełnia współrzędne statku

:param x1: współrzędna x początku statku

:param y1: współrzędna y początku statku  
:param x2: współrzędna x końca statku  
:param y2: współrzędna y końca statku

#### Funkcje:

##### [set\\_drown\\_state](#)

Ustawia stan statku na zatopiony  
:param name: stan

##### [get\\_taken\\_fields](#)

Zwraca zbiór współrzędnych zajętych dla danego statku

##### [add\\_ship\\_coordinate](#)

Dodaje współrzędne statku do listy ship\_position  
:param x: współrzędna x pola  
:param y: współrzędna y pola

#### Klasa [Competitor](#)

Klasa reprezentująca uczestnika bitwy

##### [Konstruktor](#)

konstruktor  
:param name: nazwa uczestnika

#### Funkcje:

##### [place\\_computer\\_ships](#)

Rozstawia losowo statki komputera

##### [try\\_place\\_ship](#)

Próba umieszczenia statku  
:param x1: współrzędna x początku statku  
:param y1: współrzędna y początku statku  
:param x2: współrzędna x końca statku  
:param y2: współrzędna y końca statku

##### [reset](#)

Resetuje atrybuty, zmienne klasy

##### [get\\_competitor\\_name](#)

Zwraca zmienną zawierającą nazwę uczestnika

##### [get\\_ships\\_list](#)

Zwraca listę statków umieszczonych na planszy uczestnika

##### [get\\_ships\\_to\\_place](#)

Zwraca statki do rozmieszczenia

##### [get\\_my\\_shots](#)

Zwraca set strzały uczestnika

#### [ships\\_to\\_place\\_info](#)

Zwraca informację o aktualnych statkach do rozstawienia

#### [set\\_ships\\_to\\_place](#)

Przypisuje do zmiennej ships\_to\_place listę okrętów do rozstawienia znajdującą się w res.py

#### [take\\_shot](#)

Dodaje do listy strzałów parę współrzędnych nowego strzału

#### [drown\\_check](#)

Sprawdza czy któryś statek nie jest zatopiony

#### [hit\\_check](#)

Sprawdza czy statek został trafiony

:param x : współrzędna x wybranego pola

:param y: współrzędna y wybranego pola

#### [determine\\_possible\\_coordinates](#)

Wybiera i zwraca możliwe współrzędne do strzału

#### [get\\_drown\\_ships\\_taken\\_fields](#)

Zwraca zbiór współrzędnych zajętych przez zatopione okręty

#### [ship\\_detection](#)

Dodaje statek do listy \_\_hit\_ships, lub gdy już się tam znajduje dodaje współrzędne

:param x: współrzędna x strzelonego pola

:param y: współrzędna t strzelonego pola

:param name: stan statku

## Moduł res

#### Lista [LIST\\_OF\\_SHIPS](#)

Lista zawierająca statki do rozmieszczenia, może być zmieniana

#### Klasa [Strings](#)

Klasa zawierająca wszystkie napisy

#### Klasa [States](#)

Klasa zawierająca stany przycisków

#### Klasa [Exceptions](#)

Klasa zawierająca zdefiniowany wyjątki

#### Klasa [Colors](#)

Klasa zawierająca zdefiniowany kolory

## Moduł [test module](#)

### Klasa [Tests](#)

Klasa zawierający wszystkie testy

#### Funkcje:

##### [place ships properly and start game](#)

Funkcja pomocnicza, rozstawia poprawnie statki i rozpoczyna grę

##### [test place ships improperly](#)

Próba niepoprawnego ustawienia okrętu (stykanie się bokami lub rogami). Oczekiwana informacja o błędzie

##### [test place ships properly and start game](#)

Poprawne rozmieszczenie wszystkich okrętów przez gracza i wciśnięcie przycisku rozpoczęcia gry.

##### [test empty field shoot](#)

Strzelanie w puste pole.

##### [test ship hit](#)

Trafienie w okręt przeciwnika.

##### [test try shoot own ship](#)

Próba zestrzelenia swojego okrętu - oczekiwane niepowodzenie.

##### [test double empty field shoot](#)

Próba ponownego strzelania w puste pole - oczekiwane niepowodzenie.

##### [test double ship hit](#)

Próba ponownego strzelania w okręt przeciwnika - oczekiwane niepowodzenie.

##### [test place some ships and reset](#)

Rozmieszczenie części okrętów, wciśnięcie przycisku reset - oczekiwany reset plansz.

##### [test take some shots then reset and again take some shots](#)

Poprawne rozmieszczenie wszystkich okrętów, oddanie kilku strzałów, rozpoczęcie nowej gry, ponowne poprawne rozmieszczenie okrętów, oddanie strzałów w te same pola.

##### [test win game](#)

Wygranie gry (np. Przez pokazanie okrętów przeciwnika). Rozpoczęcie nowej gry bez ponownego uruchamiania programu.

##### [test lose game](#)

Przegranie gry (np. Przez aktywację super-instynktu gracza komputera). Rozpoczęcie nowej gry bez ponownego uruchamiania programu.