

## Projekt 2.6

Piotr Matysiak

PRI  
grupa 1AR



# Contents

<b>1</b>	<b>Data Structure Index</b>	<b>1</b>
1.1	Data Structures . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Data Structure Documentation</b>	<b>5</b>
3.1	PolePlanszy Struct Reference . . . . .	5
3.1.1	Detailed Description . . . . .	5
3.1.2	Member Enumeration Documentation . . . . .	5
3.1.2.1	dozwolenie . . . . .	5
3.1.2.2	stan . . . . .	6
3.1.3	Field Documentation . . . . .	6
3.1.3.1	czyMoznaUstawicStatek . . . . .	6
3.1.3.2	obiekt . . . . .	6
<b>4</b>	<b>File Documentation</b>	<b>7</b>
4.1	znaki.c File Reference . . . . .	7
4.1.1	Macro Definition Documentation . . . . .	9
4.1.1.1	DEBUG . . . . .	9
4.1.1.2	HEIGHT . . . . .	10
4.1.1.3	KURSOR . . . . .	10
4.1.1.4	PUDLO . . . . .	10
4.1.1.5	STATEK . . . . .	10
4.1.1.6	STATEK_USTAWIANY . . . . .	10

4.1.1.7	STATUS_X_MAX . . . . .	10
4.1.1.8	STATUS_X_MIN . . . . .	11
4.1.1.9	STATUS_Y_MAX . . . . .	11
4.1.1.10	STATUS_Y_MIN . . . . .	11
4.1.1.11	TARGET_X_MAX . . . . .	11
4.1.1.12	TARGET_X_MIN . . . . .	11
4.1.1.13	TARGET_Y_MAX . . . . .	11
4.1.1.14	TARGET_Y_MIN . . . . .	12
4.1.1.15	TRAFIONY . . . . .	12
4.1.1.16	WIDTH . . . . .	12
4.1.1.17	WODA . . . . .	12
4.1.1.18	WODA_NIEDOZWOLONA . . . . .	12
4.1.1.19	ZATOPIONY . . . . .	12
4.1.2	Function Documentation . . . . .	13
4.1.2.1	czytajKlawisz() . . . . .	13
4.1.2.2	gra() . . . . .	13
4.1.2.3	inicjalizujPlansze() . . . . .	13
4.1.2.4	main() . . . . .	13
4.1.2.5	odswiezLewaPlansze() . . . . .	13
4.1.2.6	odswiezPole() . . . . .	13
4.1.2.7	ruchKursora() . . . . .	14
4.1.2.8	rysujPlansze() . . . . .	14
4.1.2.9	sprawdzMiejsce() . . . . .	14
4.1.2.10	sprawdzMiejsceAI() . . . . .	15
4.1.2.11	strzalAI() . . . . .	15
4.1.2.12	strzalGracza() . . . . .	15
4.1.2.13	turaAI() . . . . .	16
4.1.2.14	turaGracza() . . . . .	16
4.1.2.15	ustawKolory() . . . . .	16
4.1.2.16	ustawOkret() . . . . .	16

4.1.2.17	ustawOkretAI()	17
4.1.2.18	ustawOkretyAI()	17
4.1.2.19	ustawOkretyGracza()	17
4.1.2.20	wyswietlSciage()	17
4.1.2.21	zmienStan()	17
4.1.2.22	zmienStanAI()	18
4.1.3	Variable Documentation	18
4.1.3.1	kratkiOkretowAI	18
4.1.3.2	kratkiOkretowGracza	18
4.1.3.3	planszaAI	19
4.1.3.4	planszaGracza	19
4.1.3.5	x_cur	19
4.1.3.6	y_cur	19
4.1.3.7	zwyciestwo	19
<b>Index</b>		<b>21</b>



# Chapter 1

## Data Structure Index

### 1.1 Data Structures

Here are the data structures with brief descriptions:

<b>PolePlanszy</b>	
Pojedyncze pole planszy . . . . .	5





## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

<b>znaki.c</b> . . . . .	7
--------------------------	---



## Chapter 3

# Data Structure Documentation

### 3.1 PolePlanszy Struct Reference

Pojedyncze pole planszy.

#### Public Types

- enum **stan** {  
    **woda** = WODA, **statek\_ustawiany** = STATEK\_USTAWIANY, **statek** = STATEK, **trafiony** = TRAFIONY,  
    **zatopiony** = ZATOPIONY, **pudlo** = PUDLO }  
    *obiekt umieszczony w polu planszy*
- enum **dozwozenie** { **niedozwolony** = 0, **dozwolony** = 1 }  
    *okreslenie, czy mozna na tym polu ustawic statek w poczatkowej fazie gry*

#### Data Fields

- enum **PolePlanszy::stan** **obiekt**
- enum **PolePlanszy::dozwozenie** **czyMoznaUstawicStatek**

#### 3.1.1 Detailed Description

Pojedyncze pole planszy.

#### 3.1.2 Member Enumeration Documentation

##### 3.1.2.1 dozwozenie

enum **dozwozenie**

okreslenie, czy mozna na tym polu ustawic statek w poczatkowej fazie gry

**Enumerator**

niedozwolony	
dozwolony	

**3.1.2.2 stan**

enum **stan**

obiekt umieszczony w polu planszy

**Enumerator**

woda	
statek_ustawiany	
statek	
trafiony	
zatopiony	
pudlo	

**3.1.3 Field Documentation****3.1.3.1 czyMoznaUstawicStatek**

enum **PolePlanszy::dozwolenie** czyMoznaUstawicStatek

**3.1.3.2 obiekt**

enum **PolePlanszy::stan** obiekt

The documentation for this struct was generated from the following file:

- **znaki.c**

## Chapter 4

# File Documentation

### 4.1 znaki.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <ncurses.h>
```

#### Data Structures

- struct **PolePlanszy**  
*Pojedyncze pole planszy.*

#### Macros

- #define **DEBUG** 0  
*Podstawy programowania - projekt 2.6 - Program realizujący grę „Okrety” w wersji dla jednego gracza. Oponentem ma być „komputer-program”.*
- #define **WIDTH** 80  
*szerokosc terminala*
- #define **HEIGHT** 24  
*wysokosc terminala*
- #define **TARGET\_X\_MIN** 17  
*pozycja lewej krawedzi lewej planszy*
- #define **TARGET\_X\_MAX** 17 + 18  
*pozycja prawe krawedzi lewej planszy*
- #define **TARGET\_Y\_MIN** 7  
*pozycja gornej krawedzi lewej planszy*
- #define **TARGET\_Y\_MAX** 7 + 9  
*pozycja dolnej krawedzi lewej planszy*
- #define **STATUS\_X\_MIN** 43  
*pozycja lewej krawedzi prawej planszy*
- #define **STATUS\_X\_MAX** 43 + 18  
*pozycja prawej krawedzi prawej planszy*

- **#define STATUS\_Y\_MIN 7**  
*pozycja gornej krawedzi prawej planszy*
- **#define STATUS\_Y\_MAX 7 + 9**  
*pozycja dolnej krawedzi prawej planszy*
- **#define WODA 1**  
*obiekt woda*
- **#define STATEK\_USTAWIANY 2**  
*obiekt statek ustawiany w poczatkowej fazie gry*
- **#define STATEK 3**  
*obiekt statek*
- **#define TRAFIONY 4**  
*obiekt statek trafiony*
- **#define ZATOPIONY 5**  
*obiekt statek zatopiony - nieuzywany w tej wersji*
- **#define PUDLO 6**  
*obiekt puste pole trafione*
- **#define WODA\_NIEDOZWOLONA 7**  
*wyglad wody na ktorej nie mozna postawic okretu*
- **#define KURSOR 8**  
*wyglad kursora*

## Functions

- void **ustawKolory ()**  
*Funkcja ustawKolory ustawia pary kolorow w ncurses dla roznnych pol.*
- void **rysujPlansze ()**  
*Funkcja rysujPlansze rysuje na ekranie plansze z ramkami, literami i cyframi i wypelnia je woda.*
- void **odswiezLewaPlansze ()**  
*Funkcja odswiezLewaPlansze wywoluje funkcje odswiezPole dla kazdego pola lewej planszy.*
- void **zmienStan** (int x\_curs, int y\_curs, int stan)
- void **odswiezPole** (int x\_curs, int y\_curs)
- void **ruchKursora** (int key)
- void **wyswietlSciage ()**  
*Funkcja wyswietlSciage wypisuje na dole ekranu przydatne informacje.*
- void **czytajKlawisz ()**  
*Funkcja czytajKlawisz czeka na naciśnięcie spacji aby wyjść z gry.*
- void **inicjalizujPlansze ()**  
*Funkcja inicjalizujPlansze wypelnia tablice danych gracza i komputera woda.*
- void **ustawOkretyGracza ()**  
*Funkcja ustawOkretyGracza wywoluje funkcje ustawOkret dla czteromasztowca, dwoch trojmasztowcow, trzech dwumasztowcow i czterech jednomasztowcow gracza.*
- void **ustawOkret** (int wielkoscOkretu)
- int **sprawdzMiejsce** (int x\_curs, int y\_curs)
- void **gra ()**  
*Funkcja gra wywoluje po kolei funkcje turaGracza i turaAI az do trafienia wszystkich statkow przez gracza lub komputer. Na koniec wyswietla stosowny komunikat i wywoluje funkcje czytajKlawisz.*
- void **turaGracza ()**  
*Funkcja turaGracza obsluguje ruch kursora przy ataku, wywoluje funkcje strzalGracza po naciśnięciu spacji tak dlugo jak trafiany jest statek lub uprzednio trafione pole oraz pozwala na zaznaczanie pol, ktore gracz uwaza za pewne pudla X-em i rysowanie wody w wypadku pomyłki.*
- void **turaAI ()**

*Funkcja turaAI wywołuje funkcje strzalAI tak długo jak trafiony zostaje statek.*

- int **strzalGracza** (int x\_curs, int y\_curs)
- int **strzalAI** ()
- int **sprawdzMiejsceAI** (int x, int y)
- void **ustawOkretyAI** ()

*Funkcja ustawOkretyAI wywołuje funkcje ustawOkretAI dla czteromasztowca, dwóch trojmasztowcow, trzech dwumasztowcow i czterech jednomasztowcow komputera.*

- void **ustawOkretAI** (int wielkoscOkretu)
- void **zmienStanAI** (int x, int y, int stan)
- int **main** ()

*Funkcja glowna gry.*

## Variables

- struct **PolePlanszy planszaGracza** [10][10]  
*tablica dwuwymiarowa pol planczy gracza*
- struct **PolePlanszy planszaAI** [10][10]  
*tablica dwuwymiarowa pol planczy komputera*
- int **kratkiOkretowGracza**  
*liczba pol na ktorych sa okrety gracza.*
- int **kratkiOkretowAI**  
*liczba pol na ktorych sa okrety komputera.*
- int **x\_cur**  
*wspolrzedna x kursora.*
- int **y\_cur**  
*wspolrzedna y kursora.*
- int **zwyciestwo**  
*wartosc 0 dla trwajacej gry, 1 po zwyciestwie gracza i -1 po zwyciestwie komputera.*

### 4.1.1 Macro Definition Documentation

#### 4.1.1.1 DEBUG

```
#define DEBUG 0
```

Podstawy programowania - projekt 2.6 - Program realizujący grę „Okrety” w wersji dla jednego gracza. Oponentem ma być „komputer-program”.

Zasady gry: Gracz rozstawia na planszy o rozmiarach 10x10 następujące okręty: czterjednomasztowce (zajmujące jedną pozycję na planszy), trzy dwumasztowce (zajmujące dwa miejsca na planszy), dwa trzymasztowce (zajmujące trzy miejsca na planszy), jeden czteromasztowiec (zajmujący cztery miejsca na planszy). Okrety nie mogą się za sobą stykać w żaden sposób – również narożnikami pól, na których się znajdują. Oponent rozstawia swoje okręty losowo, zgodnie z powyższymi zasadami. Gracze naprzemiennie „strzelają” w planszę przeciwnika aż do zatopienia wszystkich jego okrętów. Po oddaniu strzału celnego gracz ma prawo strzelać dalej aż do chybienia.

#### Author

Piotr Matysiak - grupa 1AR

#### Date

2017-2018zmienna do testowania

#### 4.1.1.2 HEIGHT

```
#define HEIGHT 24
```

wysokosc terminala

#### 4.1.1.3 KURSOR

```
#define KURSOR 8
```

wyglad kursora

#### 4.1.1.4 PUDLO

```
#define PUDLO 6
```

obiekt puste pole trafione

#### 4.1.1.5 STATEK

```
#define STATEK 3
```

obiekt statek

#### 4.1.1.6 STATEK\_USTAWIANY

```
#define STATEK_USTAWIANY 2
```

obiekt statek ustawiany w poczatkowej fazie gry

#### 4.1.1.7 STATUS\_X\_MAX

```
#define STATUS_X_MAX 43 + 18
```

pozycja prawej krawedzi prawej planszy



#### 4.1.1.8 STATUS\_X\_MIN

```
#define STATUS_X_MIN 43
```

pozycja lewej krawedzi prawej planszy

#### 4.1.1.9 STATUS\_Y\_MAX

```
#define STATUS_Y_MAX 7 + 9
```

pozycja dolnej krawedzi prawej planszy

#### 4.1.1.10 STATUS\_Y\_MIN

```
#define STATUS_Y_MIN 7
```

pozycja gornej krawedzi prawej planszy

#### 4.1.1.11 TARGET\_X\_MAX

```
#define TARGET_X_MAX 17 + 18
```

pozycja prawe krawedzi lewej planszy

#### 4.1.1.12 TARGET\_X\_MIN

```
#define TARGET_X_MIN 17
```

pozycja lewej krawedzi lewej planszy

#### 4.1.1.13 TARGET\_Y\_MAX

```
#define TARGET_Y_MAX 7 + 9
```

pozycja dolnej krawedzi lewej planszy

#### 4.1.1.14 TARGET\_Y\_MIN

```
#define TARGET_Y_MIN 7
```

pozycja gornej krawedzi lewej planszy

#### 4.1.1.15 TRAFIONY

```
#define TRAFIONY 4
```

obiekt statek trafiony

#### 4.1.1.16 WIDTH

```
#define WIDTH 80
```

szerokosc terminala

#### 4.1.1.17 WODA

```
#define WODA 1
```

obiekt woda

#### 4.1.1.18 WODA\_NIEDOZWOLONA

```
#define WODA_NIEDOZWOLONA 7
```

wyglad wody na ktorej nie mozna postawic okretu

#### 4.1.1.19 ZATOPIONY

```
#define ZATOPIONY 5
```

obiekt statek zatopiony - nieuzywany w tej wersji

## 4.1.2 Function Documentation

### 4.1.2.1 czytajKlawisz()

```
void czytajKlawisz ( )
```

Funkcja czytajKlawisz czeka na naciśnięcie spacji aby wyjść z gry.

### 4.1.2.2 gra()

```
void gra ( )
```

Funkcja gra wywołuje po kolei funkcje turaGracza i turaAI aż do trafienia wszystkich statków przez gracza lub komputer. Na koniec wyświetla stosowny komunikat i wywołuje funkcję czytajKlawisz.

### 4.1.2.3 inicjalizujPlansze()

```
void inicjalizujPlansze ( )
```

Funkcja inicjalizujPlansze wypełnia tablice danych gracza i komputera wodą.

### 4.1.2.4 main()

```
int main ( )
```

Funkcja główna gry.

### 4.1.2.5 odswiezLewaPlansze()

```
void odswiezLewaPlansze ( )
```

Funkcja odswiezLewaPlansze wywołuje funkcję odswiezPole dla każdego pola lewej planszy.

### 4.1.2.6 odswiezPole()

```
void odswiezPole (
    int x_curs,
    int y_curs )
```

Funkcja odswiezPole rysuje pojedyncze pole na planszy zgodnie z zawartością tabeli danych.

**Parameters**

in	<i>x_curs</i>	wspolrzedna x polozenia kursora
in	<i>y_curs</i>	wspolrzedna y polozenia kursora

**4.1.2.7 ruchKursora()**

```
void ruchKursora (
    int key )
```

Funkcja ruchKursora obsluguje ruch kursora po naciskaniu strzalek na klawiaturze.

**Parameters**

in	<i>key</i>	wartosc liczbowa naciśniętego klawisza.
----	------------	---

**4.1.2.8 rysujPlansze()**

```
void rysujPlansze ( )
```

Funkcja rysujPlansze rysuje na ekranie plansze z ramkami, literami i cyframi i wypelnia je woda.

**4.1.2.9 sprawdzMiejsce()**

```
int sprawdzMiejsce (
    int x_curs,
    int y_curs )
```

Funkcja sprawdzMiejsce sprawdza czy dane pole tablicy ma wartosc czyMoznaUstawicStatek == dozwolony.

**Parameters**

in	<i>x_curs</i>	wspolrzedna x polozenia kursora
in	<i>y_curs</i>	wspolrzedna y polozenia kursora

**Returns**

wartosc 1 jezeli ruch jest legalny, i -1 jezeli nie jest.

#### 4.1.2.10 sprawdzMiejsceAI()

```
int sprawdzMiejsceAI (
    int x,
    int y )
```

Funkcja sprawdzMiejsceAI sprawdza czy dane pole tablicy ma wartosc czyMoznaUstawicStatek == dozwolony.

##### Parameters

in	x	wspolrzedna x trafienia komputera
in	y	wspolrzedna y trafienia komputera

##### Returns

wartosc 1 jezeli ruch jest legalny, i -1 jezeli nie jest.

#### 4.1.2.11 strzalAI()

```
int strzalAI ( )
```

Funkcja strzalAI wybiera za pomoca generatora liczb pseudolosowych pole do trafienia przez komputer, sprawdza jaki jest stan trafionego pola, zmienia ten stan w odpowiedni sposob i wywołuje funkcje odswiezPole. Wypisuje na gorze ekranu liczbe pozostalych pol do trafienia. Funkcja zmienia zmienna globalna zwyciestwo na -1 jezeli zostaly trafione wszystkie okrety gracza.

##### Returns

wartosc 0 jezeli zostal trafiony statek lub uprzednio trafione pole, i 1 jezeli komputer spudlowal.

#### 4.1.2.12 strzalGracza()

```
int strzalGracza (
    int x_curs,
    int y_curs )
```

Funkcja strzalGracza sprawdza jaki jest stan trafionego pola, zmienia ten stan w odpowiedni sposob i wywołuje funkcje odswiezPole. Wypisuje na gorze ekranu liczbe pozostalych pol do trafienia. Funkcja zmienia zmienna globalna zwyciestwo na 1 jezeli zostaly trafione wszystkie okrety komputera.

##### Parameters

in	x_curs	wspolrzedna x polozenia kursora
in	y_curs	wspolrzedna y polozenia kursora

### Returns

wartosc 1 jezeli zostal trafiony statek lub uprzednio trafione pole, i 0 jezeli gracz spudlowal.

#### 4.1.2.13 turaAI()

```
void turaAI ( )
```

Funkcja turaAI wywołuje funkcję strzałAI tak długo jak trafiony zostaje statek.

#### 4.1.2.14 turaGracza()

```
void turaGracza ( )
```

Funkcja turaGracza obsługuje ruch kursora przy ataku, wywołuje funkcję strzałGracza po naciśnięciu spacji tak długo jak trafiany jest statek lub uprzednio trafione pole oraz pozwala na zaznaczanie pól, które gracz uważa za pewne pudła X-em i rysowanie wody w wypadku pomyłki.

#### 4.1.2.15 ustawKolory()

```
void ustawKolory ( )
```

Funkcja ustawKolory ustawia pary kolorów w ncurses dla różnych pól.

#### 4.1.2.16 ustawOkret()

```
void ustawOkret (
    int wielkoscOkretu )
```

Funkcja ustawOkret próbuje ustawić na planszy okret danej wielkości w miejscu określonym przez gracza. Aby sprawdzić legalność ruchu wywołuje funkcję sprawdzMiejsce, a następnie wywołuje funkcję zmienStan dla określonych pól. Na koniec wywołuje funkcję odświeżLewaPlansze.

### Parameters

in	<i>wielkoscOkretu</i>	długość okretu równa 1, 2, 3 lub 4 pola.
----	-----------------------	--

#### 4.1.2.17 ustawOkretAI()

```
void ustawOkretAI (
    int wielkoscOkretu )
```

Funkcja ustawOkretAI probuje ustawic na planszy okret danej wielkosci w miejscu wylosowanym przez generator liczb pseudolosowych. Aby sprawdzic legalnosc ruchu wywoluje funkcje sprawdzMiejsceAI, a nastepnie wywoluje funkcje zmienStanAI dla okreslonych pol.

##### Parameters

in	wielkoscOkretu	dlugosc okretu rowna 1, 2, 3 lub 4 pola.
----	----------------	--

#### 4.1.2.18 ustawOkretyAI()

```
void ustawOkretyAI ( )
```

Funkcja ustawOkretyAI wywoluje funkcje ustawOkretAI dla czteromasztowca, dwoch trojmasztowcow, trzech dwumasztowcow i czterech jednomasztowcow komputera.

#### 4.1.2.19 ustawOkretyGracza()

```
void ustawOkretyGracza ( )
```

Funkcja ustawOkretyGracza wywoluje funkcje ustawOkret dla czteromasztowca, dwoch trojmasztowcow, trzech dwumasztowcow i czterech jednomasztowcow gracza.

#### 4.1.2.20 wyswietlSciage()

```
void wyswietlSciage ( )
```

Funkcja wyswietlSciage wypisuje na dole ekranu przydatne informacje.

#### 4.1.2.21 zmienStan()

```
void zmienStan (
    int x_curs,
    int y_curs,
    int stan )
```

Funkcja zmienStan przyporadkowuje polu odpowiedni obiekt: statek, statek ustawiany, statek trafiony, wode lub pudlo i ustawia legalnosc postawienia na polu statku.

**Parameters**

in	<i>x_curs</i>	wspolrzedna x polozenia kursora
in	<i>y_curs</i>	wspolrzedna y polozenia kursora
in	<i>stan</i>	docelowo stan pola

**4.1.2.22 zmienStanAI()**

```
void zmienStanAI (  
    int x,  
    int y,  
    int stan )
```

Funkcja zmienStanAI przyporządkowuje polu odpowiedni obiekt: statek, statek ustawiany, statek trafiony, wode lub pudło i ustawia legalność postawienia na polu statku.

**Parameters**

in	<i>x</i>	wspolrzedna x trafienia komputera
in	<i>y</i>	wspolrzedna y trafienia komputera
in	<i>stan</i>	docelowo stan pola

**4.1.3 Variable Documentation****4.1.3.1 kratkiOkretowAI**

```
int kratkiOkretowAI
```

liczba pol na ktorych sa okrety komputera.

**4.1.3.2 kratkiOkretowGracza**

```
int kratkiOkretowGracza
```

liczba pol na ktorych sa okrety gracza.



#### 4.1.3.3 planszaAI

```
struct PolePlanszy planszaAI[10][10]
```

tablica dwuwymiarowa pol planczy komputera

#### 4.1.3.4 planszaGracza

```
struct PolePlanszy planszaGracza[10][10]
```

tablica dwuwymiarowa pol planczy gracza

#### 4.1.3.5 x\_cur

```
int x_cur
```

wspolrzedna x kursora.

#### 4.1.3.6 y\_cur

```
int y_cur
```

wspolrzedna y kursora.

#### 4.1.3.7 zwyciestwo

```
int zwyciestwo
```

wartosc 0 dla trwajacej gry, 1 po zwyciestwie gracza i -1 po zwyciestwie komputera.



# Index

czyMoznaUstawicStatek  
    PolePlanszy, 6  
czytajKlawisz  
    znaki.c, 13

DEBUG  
    znaki.c, 9  
dozwolenie  
    PolePlanszy, 5

gra  
    znaki.c, 13

HEIGHT  
    znaki.c, 9

inicjalizujPlansze  
    znaki.c, 13

KURSOR  
    znaki.c, 10  
kratkiOkretowAI  
    znaki.c, 18  
kratkiOkretowGracza  
    znaki.c, 18

main  
    znaki.c, 13

obiekt  
    PolePlanszy, 6  
odswiezLewaPlansze  
    znaki.c, 13  
odswiezPole  
    znaki.c, 13

PUDLO  
    znaki.c, 10  
planszaAI  
    znaki.c, 18  
planszaGracza  
    znaki.c, 19  
PolePlanszy, 5  
    czyMoznaUstawicStatek, 6  
    dozwolenie, 5  
    obiekt, 6  
    stan, 6

ruchKursora  
    znaki.c, 14  
rysujPlansze  
    znaki.c, 14

STATEK\_USTAWIANY  
    znaki.c, 10  
STATEK  
    znaki.c, 10  
STATUS\_X\_MAX  
    znaki.c, 10  
STATUS\_X\_MIN  
    znaki.c, 10  
STATUS\_Y\_MAX  
    znaki.c, 11  
STATUS\_Y\_MIN  
    znaki.c, 11  
sprawdzMiejsce  
    znaki.c, 14  
sprawdzMiejsceAI  
    znaki.c, 14  
stan  
    PolePlanszy, 6  
strzalAI  
    znaki.c, 15  
strzalGracza  
    znaki.c, 15

TARGET\_X\_MAX  
    znaki.c, 11  
TARGET\_X\_MIN  
    znaki.c, 11  
TARGET\_Y\_MAX  
    znaki.c, 11  
TARGET\_Y\_MIN  
    znaki.c, 11  
TRAFIONY  
    znaki.c, 12  
turaAI  
    znaki.c, 16  
turaGracza  
    znaki.c, 16

ustawKolory  
    znaki.c, 16  
ustawOkret  
    znaki.c, 16  
ustawOkretAI  
    znaki.c, 16  
ustawOkretyAI  
    znaki.c, 17  
ustawOkretyGracza  
    znaki.c, 17

WIDTH  
    znaki.c, 12  
WODA\_NIEDOZWOLONA  
    znaki.c, 12  
WODA  
    znaki.c, 12  
wyswietlSciage  
    znaki.c, 17  
  
x\_cur  
    znaki.c, 19  
  
y\_cur  
    znaki.c, 19  
  
ZATOPIONY  
    znaki.c, 12  
zmienStan  
    znaki.c, 17  
zmienStanAI  
    znaki.c, 18  
znaki.c, 7  
    czytajKlawisz, 13  
    DEBUG, 9  
    gra, 13  
    HEIGHT, 9  
    inicjalizujPlansze, 13  
    KURSOR, 10  
    kratkiOkretowAI, 18  
    kratkiOkretowGracza, 18  
    main, 13  
    odswiezLewaPlansze, 13  
    odswiezPole, 13  
    PUDLO, 10  
    planszaAI, 18  
    planszaGracza, 19  
    ruchKursora, 14  
    rysujPlansze, 14  
    STATEK\_USTAWIANY, 10  
    STATEK, 10  
    STATUS\_X\_MAX, 10  
    STATUS\_X\_MIN, 10  
    STATUS\_Y\_MAX, 11  
    STATUS\_Y\_MIN, 11  
    sprawdzMiejsce, 14  
    sprawdzMiejsceAI, 14  
    strzalAI, 15  
    strzalGracza, 15  
    TARGET\_X\_MAX, 11  
    TARGET\_X\_MIN, 11  
    TARGET\_Y\_MAX, 11  
    TARGET\_Y\_MIN, 11  
    TRAFIONY, 12  
    turaAI, 16  
    turaGracza, 16  
    ustawKolory, 16  
    ustawOkret, 16  
    ustawOkretAI, 16  
    ustawOkretyAI, 17  
    ustawOkretyGracza, 17  
    WIDTH, 12  
    WODA\_NIEDOZWOLONA, 12  
    WODA, 12  
    wyswietlSciage, 17  
    x\_cur, 19  
    y\_cur, 19  
    ZATOPIONY, 12  
    zmienStan, 17  
    zmienStanAI, 18  
    zwyciestwo, 19  
zwyciestwo  
    znaki.c, 19