

Temat projektu: Symulator układów planetarnych

Skład grupy projektowej:

Damian Mucha - lider zespołu

Michał Sujewicz

Opis zadania symulacji w języku naturalnym:

Użytkownik tworząc nowy układ podaje nazwę układu, liczbę planet i planetoid w układzie, okres trwania symulacji. W czasie trwania symulacji może manipulować między innymi jej tempem i powiększeniem wyświetlanych obiektów. Na końcowy wynik mają wpływ zdarzenia losowe takie jak kolizje obiektów lub ewolucja gwiazdy. Program zwraca między innymi informacje o rodzajach wygenerowanych układów, które z planet skalistych są zdadne do skolonizowania przez człowieka oraz dane gazowych olbrzymów. Użytkownik może na koniec przeanalizować wyniki i zapis przebiegu symulacji lub rozpocząć symulację kolejnego układu.

Analiza czasownikowo-rzeczownikowa:

Zaprojektowaliśmy prostą symulację agentową, w której tworzymy układy planetarne i spośród wielu rodzajów planet szukamy takich, które są zdadne do skolonizowania przez człowieka. Dla uproszczenia przyjmujemy że wszystkie obiekty poruszają się w jednej płaszczyźnie. W trakcie trwania symulacji występują zdarzenia losowe mające wpływ na stan końcowy układu.

1. Zachowanie się obiektów:
 - wszystkie planety orbitują wokół gwiazdy i oddziałują na siebie wzajemnie;
 - planetoidy mają swoją prędkość początkową i mogą zderzać się z planetami;
 - kolizje z planetoidą powodują zmniejszenie współczynnika kolonizacji planety.
2. Parametry symulacji:
 - liczba analizowanych układów;
 - zakres ilości planet w układach;
 - zakres ilości planetoid w układach;
 - czas trwania symulacji.

Diagram klas

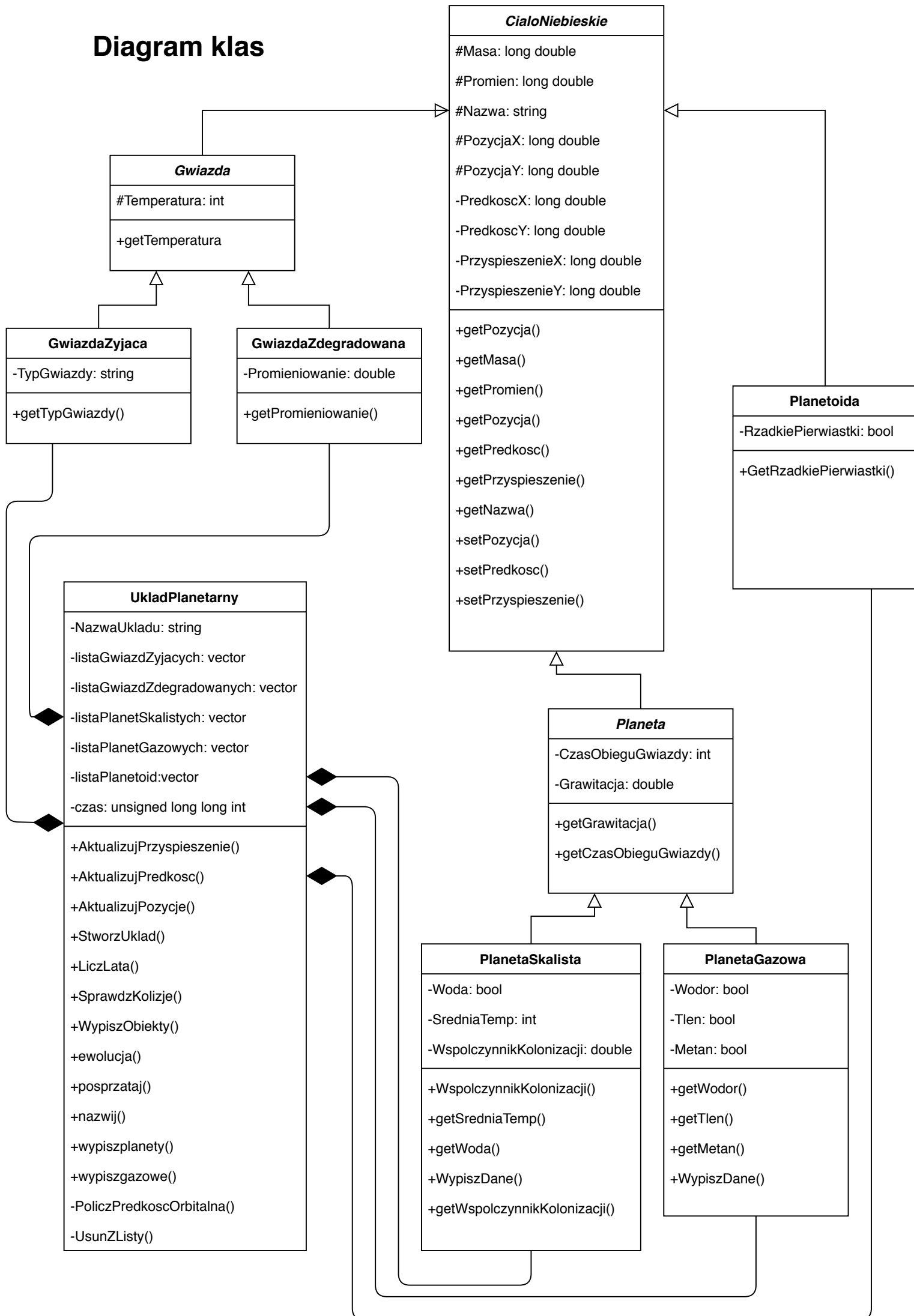


Diagram obiektów

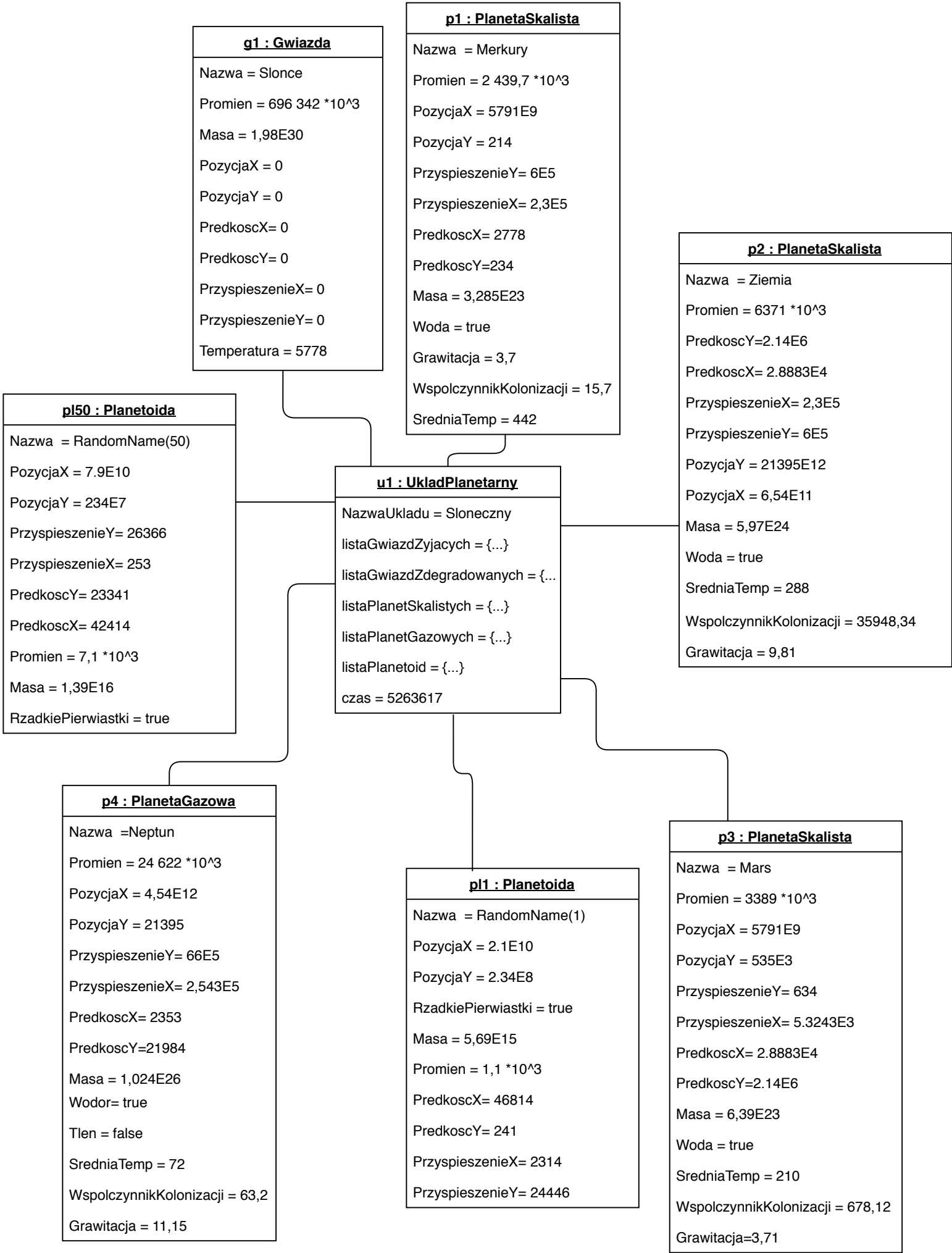
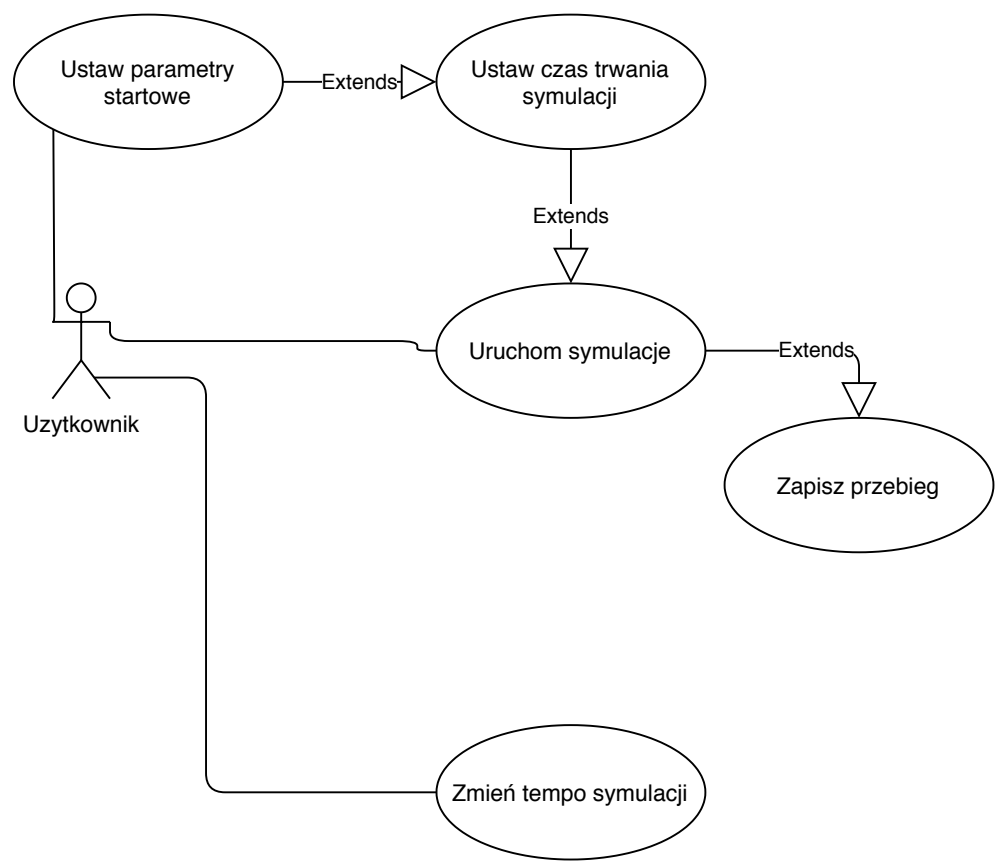


Diagram przypadków użycia



Classname: CialoNiebieskie	
Superclasses: none	
Subclass(es): Planeta, Gwiazda, Planetoida	
Responsibilities: - generuje pozycje - generuje mase - generuje promien - nazywa cialo	Collaboration: -UkladPlanetarny

Classname: Planeta	
Superclasses: CialoNiebieskie	
Subclass(es): PlanetaSkalista, PlanetaGazowa	
Responsibilities: - generuje grawitacje - generuje predkosc - generuje czas obiegu	Collaboration: - UkladPlanetarny

Classname: Gwiazda	
Superclasses: CialoNiebieskie	
Subclass(es): GwiazdaZyjaca, GwiazdaZdegradowana	
Responsibilities: - generuje temperature	Collaboration: - UkladPlanetarny

Classname: Planetoida	
Superclasses: CialoNiebieskie	
Subclass(es): none	
Responsibilities: - generuje rzadkie pierwiastki - generuje predkosc	Collaboration: UkladPlanetarny

Classname: GwiazdaZyjaca	
Superclasses: Gwiazda	
Subclass(es): none	
Responsibilities: - generuje typ gwiazdy	Collaboration: - UkladPlanetarny

Classname: GwiazdaZdegradowana	
Superclasses: Gwiazda	
Subclass(es): none	
Responsibilities: - generuje promieniowanie	Collaboration: - UkladPlanetarny

Classname: PlanetaSkalista	
Superclasses: Planeta	
Subclass(es): none	
Responsibilities: - generuje wode - generuje srednia temperature - generuje wspolczynnik kolonizacji	Collaboration: - UkladPlanetarny

Classname: PlanetaGazowa	
Superclasses: Planeta	
Subclass(es): none	
Responsibilities: - generuje ilosc wodoru - generuje ilosc tlenu - generuje ilosc metanu	Collaboration: - UkladPlanetarny

Classname: UkladPlanetarny	
Superclasses: none	
Subclass(es): none	
Responsibilities: - nazywa układ - tworzy obiekty - zapisuje przebieg symulacji - zarządza ewolucją gwiazdy	Collaboration: - CialoNiebieskie - Planeta - Gwiazda - Planetoida - GwiazdaZyjaca - GwiazdaZdegradowana - PlanetaSkalista - PlanetaGazowa

Diagram aktywności planety:

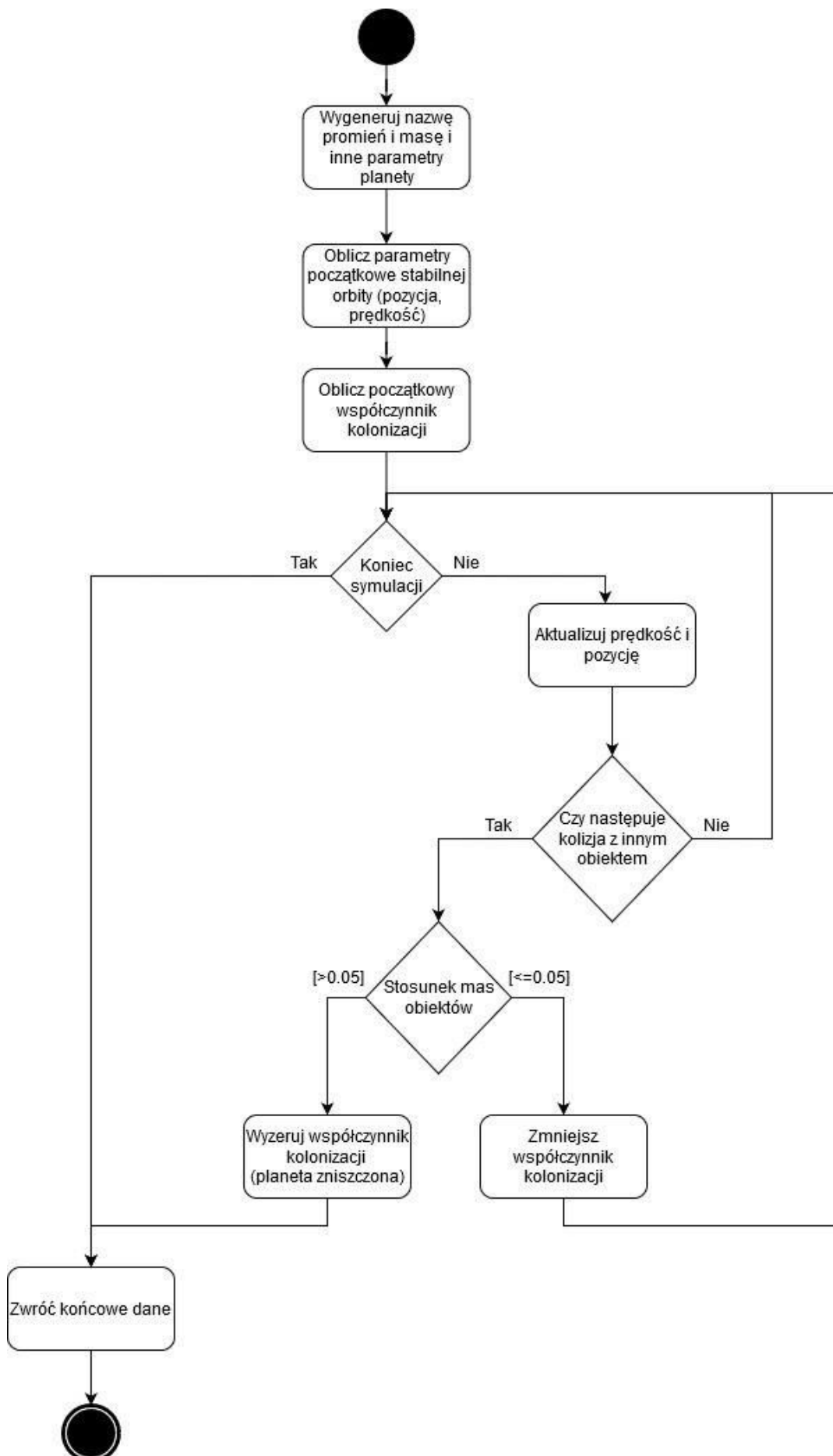
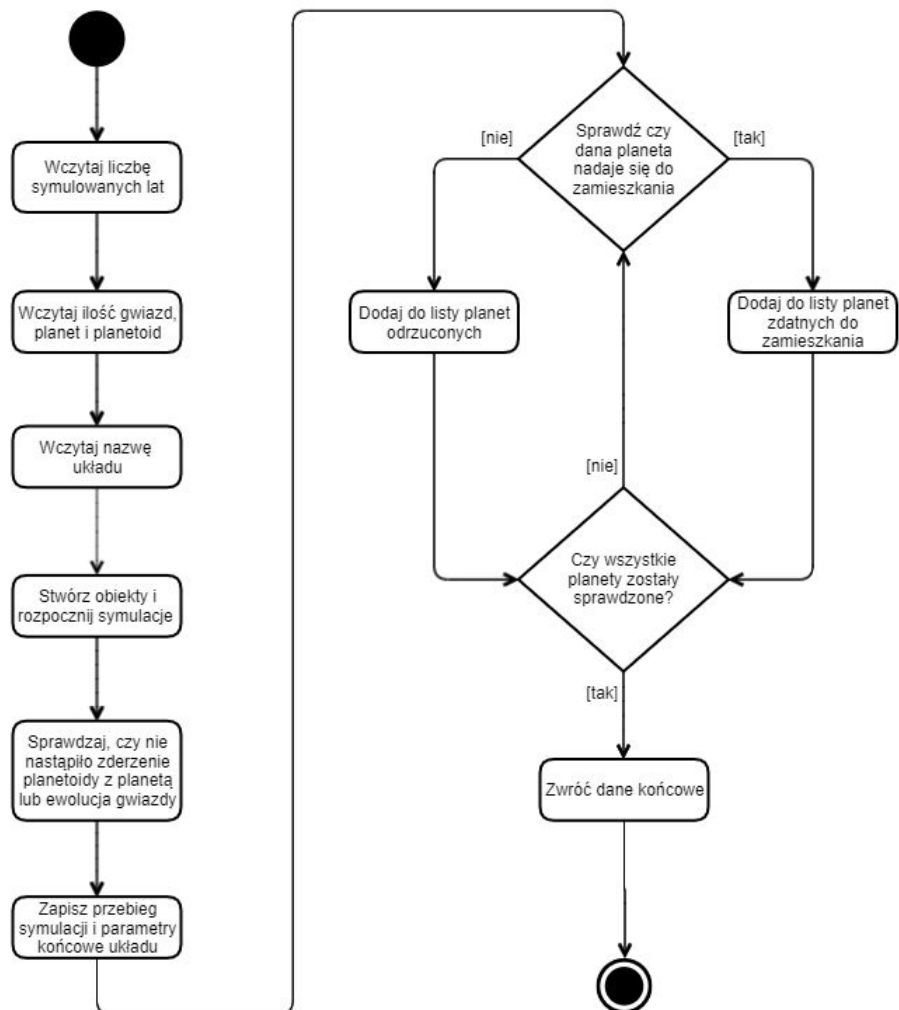


Diagram aktywności symulacji:



Solaris

Generated by Doxygen 1.8.18

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

CialoNiebieskie	??
Gwiazda	??
GwiazdaZdegradowana	??
GwiazdaZyjaca	??
Planeta	??
PlanetaGazowa	??
PlanetaSkalista	??
Planetoida	??
UkladPlanetarny	??

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

CialoNiebieskie		
	Klasa z ktorej dziedzicza niemal wszystkie pozostale	??
Gwiazda		
	Kazda gwiazda ma swoja temperature	??
GwiazdaZdegradowana		
	Gwiazda (p. ??) zdegradowana powstaje w wyniku ewolucji gwiazdy zyjacej	??
GwiazdaZyjaca		
	Gwiazda (p. ??) zyjaca zawsze powstaje w srodku ukladu	??
Planeta		
	Planeta (p. ??) posiada swoj czas obiegu gwiazdy i grawitacje	??
PlanetaGazowa		
	Planeta (p. ??) gazowa moze posiadac takie pierwiastki jak wodor, tlen czy metan	??
PlanetaSkalista		
	Planeta (p. ??) Skalista moze posiadac wode, ma swoja srednia temperature i wspolczynnik kolonizacji	??
Planetoida		
	Planetoida (p. ??) moze posiadac rzadkie pierwiastki	??
UkladPlanetarny		
	Uklad planetarny to najwieksza klasa ktora zarzadza wszystkimi obiektami i ich interakcjami . .	??

Chapter 3

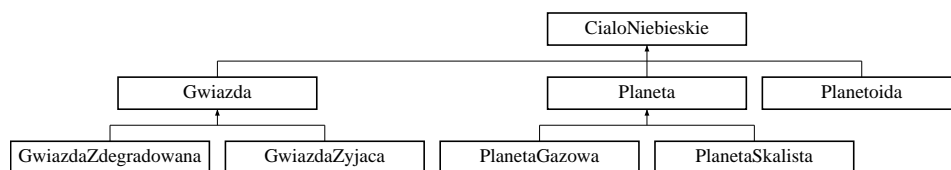
Class Documentation

3.1 CialoNiebieskie Class Reference

Klasa z ktorej dziedzicza niemal wszystkie pozostale.

```
#include <CialoNiebieskie.h>
```

Inheritance diagram for CialoNiebieskie:



Public Member Functions

- **CialoNiebieskie** (long double masa, long double promien)
- **CialoNiebieskie** (long double masa, long double promien, long double pozycjax, long double pozycjay)
- long double **getPozycjaX** ()
Funkcja zwraca pozycje X.
- long double **getPozycjaY** ()
Funkcja zwraca pozycje Y.
- long double **getMasa** ()
Funkcja zwraca mase cia³ a.
- long double **getPromien** ()
Funkcja zwraca promien cia³ a.
- long double **getPredkoscX** ()
Funkcja zwraca predkosc X.
- long double **getPredkoscY** ()
Funkcja zwraca prdkolJ Y
- long double **getPrzyspieszenieX** ()
Funkcja zwraca przyspieszenie X.
- long double **getPrzyspieszenieY** ()
Funkcja zwraca przyspieszenie Y.

- char * **getNazwa** ()
Funkcja zwraca nazwe.
- void **setPozycjaX** (long double x)
Przyjmuje wartosc i ustawia ja jako pozycje X dla ciala.
- void **setPozycjaY** (long double y)
Przyjmuje wartosc i ustawia ja jako pozycj Y dla ciala.
- void **setPrzyspieszenieX** (long double x)
Przyjmuje wartosc i ustawia ja jako przyspieszenie X dla ciala.
- void **setPrzyspieszenieY** (long double y)
Przyjmuje wartosc i ustawia ja jako przyspieszenie y dla ciala.
- void **setPredkoscX** (long double x)
Przyjmuje wartosc i ustawia ja jako predkosc X dla ciala.
- void **setPredkoscY** (long double y)
Przyjmuje wartosc i ustawia ja jako predkosc Y dla ciala.

Protected Attributes

- long double **Masa**
- long double **Promien**
- char **Nazwa** [32]
- long double **PozycjaX**
- long double **PozycjaY**

3.1.1 Detailed Description

Klasa z ktorej dziedzicza niemal wszystkie pozostale.

Kazde cialo posiada mase, promien, nazwe, pozycje X i Y, predkosc X i Y oraz przyspieszenie X i Y.

The documentation for this class was generated from the following files:

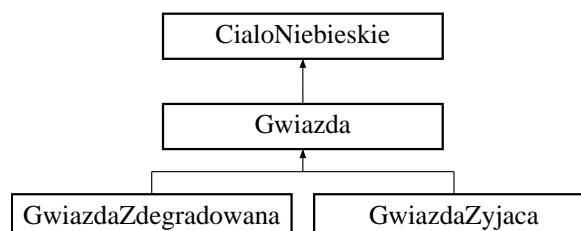
- PO-master/CialoNiebieskie.h
- PO-master/CialoNiebieskie.cpp

3.2 Gwiazda Class Reference

Kazda gwiazda ma swoja temperature.

```
#include <Gwiazda.h>
```

Inheritance diagram for Gwiazda:



Public Member Functions

- int **getTemperatura** ()
Zwraca temperature gwiazdy.

Protected Attributes

- int **Temperatura**

3.2.1 Detailed Description

Kazda gwiazda ma swoja temperature.

The documentation for this class was generated from the following files:

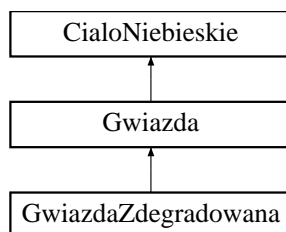
- PO-master/Gwiazda.h
- PO-master/Gwiazda.cpp

3.3 GwiazdaZdegradowana Class Reference

Gwiazda (p. ??) zdegradowana powstaje w wyniku ewolucji gwiazdy zyjacej.

```
#include <GwiazdaZdegradowana.h>
```

Inheritance diagram for GwiazdaZdegradowana:



Public Member Functions

- **GwiazdaZdegradowana** ()
- **GwiazdaZdegradowana** (long double masa, long double posX, long double posY, long double promien)
- double **getPromieniowanie** ()
Zwraca wartosc promieniowania.

Additional Inherited Members

3.3.1 Detailed Description

Gwiazda (p. ??) zdegradowana powstaje w wyniku ewolucji gwiazdy zyjacej.

Dodatkowo posiada tez promieniowanie.

3.3.2 Constructor & Destructor Documentation

3.3.2.1 GwiazdaZdegradowana()

```
GwiazdaZdegradowana::GwiazdaZdegradowana ( )
```

Konstruktor gwiazdy ustala jej pozycje, zawsze w srodku ukkladu na miejsce gwiazdy zyjacej, losuje jej promieniowanie, mase i promien.

The documentation for this class was generated from the following files:

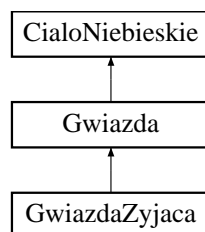
- PO-master/GwiazdaZdegradowana.h
- PO-master/GwiazdaZdegradowana.cpp

3.4 GwiazdaZyjaca Class Reference

Gwiazda (p. ??) zyjaca zawsze powstaje w srodku ukkladu.

```
#include <GwiazdaZyjaca.h>
```

Inheritance diagram for GwiazdaZyjaca:



Public Member Functions

- **GwiazdaZyjaca** ()
- char * **getTypGwiazdy** ()
Funkcja zwraca typ gwiazdy.

Additional Inherited Members

3.4.1 Detailed Description

Gwiazda (p. ??) zyjaca zawsze powstaje w srodku ukkladu.

Dodatkowo posiada tez swoj typ.

3.4.2 Constructor & Destructor Documentation

3.4.2.1 GwiazdaZyjaca()

```
GwiazdaZyjaca::GwiazdaZyjaca ( )
```

Konstruktor gwiazdy ustala jej pozycje w srodku ukkladu, generuje losowa nazwe, typ gwiazdy, oraz wlasciwa dla gwiazd zyjacych promien, mase i temeprature.

The documentation for this class was generated from the following files:

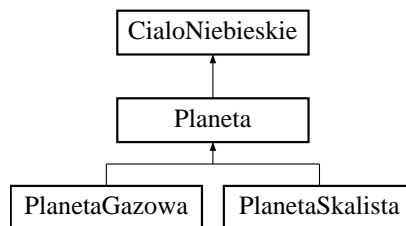
- PO-master/GwiazdaZyjaca.h
- PO-master/GwiazdaZyjaca.cpp

3.5 Planeta Class Reference

Planeta (p. ??) posiada swoj czas obiegu gwiazdy i grawitacje.

```
#include <Planeta.h>
```

Inheritance diagram for Planeta:



Public Member Functions

- double **getGrawitacja** ()
Funkcja zwraca grawitacje planety.
- int **getCzasObieguGwiazdy** ()
Funkcja zwraca czas obiegu gwiazdy planety.

Additional Inherited Members

3.5.1 Detailed Description

Planeta (p. ??) posiada swoj czas obiegu gwiazdy i grawitacje.

The documentation for this class was generated from the following files:

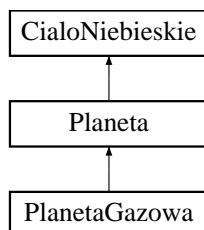
- PO-master/Planeta.h
- PO-master/Planeta.cpp

3.6 PlanetaGazowa Class Reference

Planeta (p. ??) gazowa moze posiadac takie pierwiastki jak wodor, tlen czy metan.

```
#include <PlanetaGazowa.h>
```

Inheritance diagram for PlanetaGazowa:



Public Member Functions

- void **wypiszdane** (std::ofstream &gazowe)
Wypisuje dane o planecie do pliku tekstowego.
- **PlanetaGazowa** ()
Konstruktor generujący losowa nazwe planety i wlasciwe dla planet gazowych parametry.
- bool **getWodor** ()
Funkcja zwraca informacje o tym czy na planecie jest wodor.
- bool **getTlen** ()
Funkcja zwraca informacje o tym czy na planecie jest tlen.
- bool **getMetan** ()
Funkcja zwraca informacje o tym czy na planecie jest metan.

Additional Inherited Members

3.6.1 Detailed Description

Planeta (p. ??) gazowa moze posiadac takie pierwiastki jak wodor, tlen czy metan.

The documentation for this class was generated from the following files:

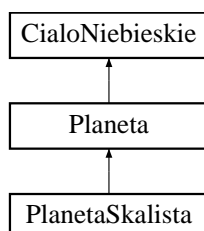
- PO-master/PlanetaGazowa.h
- PO-master/PlanetaGazowa.cpp

3.7 PlanetaSkalista Class Reference

Planeta (p. ??) Skalista moze posiadac wode, ma swoja srednia temperature i wspolczynnik kolonizacji.

```
#include <PlanetaSkalista.h>
```

Inheritance diagram for PlanetaSkalista:



Public Member Functions

- **PlanetaSkalista** ()
Konstruktor generujący losową nazwę planety i właściwe dla planet skalistych parametry.
- int **getSredniaTemp** ()
Funkcja zwraca średnią temperaturę.
- bool **getWoda** ()
Funkcja zwraca informacje o tym czy na planecie jest woda.
- double **getWspolczynnikKolonizacji** ()
Funkcja zwraca wartość współczynnika kolonizacji.
- void **wypiszdane** (std::ofstream &wynik)
Funkcja wypisuje planety zdolne do kolonizacji do pliku tekstowego.

Additional Inherited Members

3.7.1 Detailed Description

Planeta (p. ??) Skalista może posiadać wodę, ma swoją średnią temperaturę i współczynnik kolonizacji.

The documentation for this class was generated from the following files:

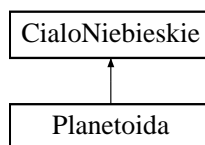
- PO-master/PlanetaSkalista.h
- PO-master/PlanetaSkalista.cpp

3.8 Planetoida Class Reference

Planetoida (p. ??) może posiadać rzadkie pierwiastki.

```
#include <Planetoida.h>
```

Inheritance diagram for Planetoida:



Public Member Functions

- **Planetoida** ()
Konstruktor nadający planetoidzie losową nazwę i właściwe dla tego typu ciała parametry.
- bool **getRzadkiePierwiastki** ()
Funkcja zwraca informacje o występowaniu rzadkich pierwiastków.

Additional Inherited Members

3.8.1 Detailed Description

Planetoida (p. ??) może posiadać rzadkie pierwiastki.

The documentation for this class was generated from the following files:

- PO-master/Planetoida.h
- PO-master/Planetoida.cpp

3.9 UkładPlanetarny Class Reference

Układ planetarny to największa klasa która zarządza wszystkimi obiektami i ich interakcjami.

```
#include <UkładPlanetarny.h>
```

Public Member Functions

- void **wypiszgazowe** (std::ofstream &gazowe)
Funkcja wypisuje dane gazowych olbrzymów do pliku tekstowego.
- void **nazwij** ()
Funkcja nazywa układ planetarny.
- void **wypiszplanety** (std::ofstream &wynik, bool gwiazda)
Funkcja wypisuje planety zdolne do kolonizacji do pliku tekstowego.
- void **AktualizujPrzyspieszenie** (std::vector< **CialoNiebieskie** * > &listaCial)
Funkcja wylicza i aktualizuje przyspieszenie dla wszystkich ciał.
- void **AktualizujPredkosc** (std::vector< **CialoNiebieskie** * > &listaCial, double Tempo)
Funkcja wylicza i aktualizuje predkosc dla wszystkich ciał.
- void **AktualizujPozycje** (std::vector< **CialoNiebieskie** * > &listaCial, double Tempo)
Funkcja wylicza i aktualizuje pozycje dla wszystkich ciał.
- void **StworzUklad** (std::vector< **CialoNiebieskie** * > *ciala, unsigned int IloscGwiazdZyjacych, unsigned int IloscPlanetSkalistych, unsigned int IloscPlanetGazowych, unsigned int IloscPlanetoid)
Funkcja inicjalizuje wszystkie obiekty.
- int **LiczLata** (double Tempo)
Funkcja przelicza czas który upłynął niezależnie od tempa symulacji.
- void **posprzataj** ()
Funkcje przed stworzeniem nowego układu czyszczy wektory z ciałami.
- void **SprawdzKolizje** (std::vector< **CialoNiebieskie** * > &listaCial, unsigned int &nObiektow, std::ofstream &zapis, int &liczbakolizji)
Funkcja sprawdza kolizje poszczególnych ciał.
- void **WypiszObiekty** (bool gwiazda)
Funkcja wypisuje ilość obiektów końcowych.
- void **ewolucja** (std::vector< **CialoNiebieskie** * > &ciala, int &kontrola, bool &gwiazda)
Funkcja zarządza ewolucją gwiazdy żyjącej i jej przemianą w gwiazdę zdegradowaną.

3.9.1 Detailed Description

Układ planetarny to największa klasa która zarządza wszystkimi obiektami i ich interakcjami.

Tworzy obiekty, rozmieszcza je w przestrzeni, oblicza ich predkosc i przyspieszenie. Posiada swoją nazwę którą wybiera użytkownik.

The documentation for this class was generated from the following files:

- PO-master/UkładPlanetarny.h
- PO-master/UkładPlanetarny.cpp