

Programowanie II R

Zadania – rozszerzenie serii 4.

Klasy

Zadanie 1: Dokładność numeryczna – Zachowanie energii i Algorytm Verleta

Całkowanie podstawową metodą Eulera nie zachowuje całkowitej energii układu. Wypisz zmiany całkowitej energii w czasie. Następnie zmień algorytm całkowania na velocity Verlet (https://en.wikipedia.org/wiki/Verlet_integration#Velocity_Verlet) i porównaj wyniki.

Zadanie 2: Eksport Danych i Wizualizacja Orbit

Zapisz położenia planet do pliku (np. w formacie csv) za pomocą `std::ofstream`. Użyj tych danych do wizualizacji ruchu planet (np. Python matplotlib).

Zadanie 3: Efekt Asysty Grawitacyjnej

1. Dodajcie do symulacji obiekt Sonda o małej masie (np. 1000 kg).
2. Umieście sondę blisko Ziemi i nadajcie jej prędkość początkową w stronę Jowisza (wymaga metody prób i błędów!).
3. Zarejestrujcie moment przelotu blisko Jowisza i udowodnijcie na podstawie danych, że prędkość końcowa sondy względem Słońca znacznie wzrosła.

Rozszerzenie 4: Przestrzeń 3D i orbita Plutona

Orbita Plutona jest nachylona względem płaszczyzny ekliptyki. Dodanie Plutona do symulacji wymaga rozszerzenia jej do trzech wymiarów.

1. Zmodyfikujcie lub utwórzcie nową klasę `Vector3d`, dodając składową `Z`. Zaktualizujcie klasę `Planeta`.
2. Dodajcie Plutona na osi `X` w odległości $29.7AU$ (peryhelium, zakładamy początkową koniunkcję).
3. Inklinację Plutona (17°) uwzględnijcie nadając mu niezerową prędkość początkową w kierunku prostopadłym do płaszczyzny ekliptyki. Prędkość orbitalna Plutona w peryhelium wynosi ok. 6.1 km/s.

Opracowanie: Piotr Dziekan.