



OPIS ZADANIA

Implementacja aplikacji klienckiej korzystającej z komponentu *DataMonitor*, który znajduje się na serwerze aplikacyjnym. Zadaniem aplikacji jest pobranie danych wejściowych w postaci liczb reprezentujących współczynniki pewnej prostej w przestrzeni 3D. Kolejne liczby w strumieniu wejściowym grupowane są w czwórki, z których każda opisuje położenie masy punktowej w przestrzeni 3D. Każda czwórka zawiera współrzędne punktu w przestrzeni oraz wartość umieszczoną w tym punkcie masy. Aplikacja oblicza moment bezwładności opisanego danymi wejściowymi układu punktów materialnych względem zdefiniowanej wcześniej osi obrotu prostej wyznaczonej przez układ równań $ax + b/2y = 0$ i $cz + b/2y = 0$.

OPIS MECHANIZMU WYSZUKIWANIA

Do zlokalizowania klasy *DataMonitor* oraz nawiązania z nią połączenie wykorzystany został interfejs JNDI (ang. Java Naming and Directory Interface). Pozwala on klientom na odkrywanie oraz wyszukiwania danych i obiektów na podstawie nazw.

Aby nawiązać połączenie należy utworzyć obiekt *InitialContext*:

```
InitialContext ctx = new InitialContext();
```

Następnie należy skorzystać z metody *lookup*, której implementacja umożliwia odnalezienie zasobu po nazwie:

```
IDataMonitor obj = (IDataMonitor)ctx.lookup("java:global/ejb-project/DataMonitor!pl.jrj.data.IDataMonitor");
```

Ścieżka przestrzeni nazw ma postać:

```
java:global/ejb-project/DataMonitor!pl.jrj.data.IDataMonitor
```

ZASTOSOWANY ALGORYTM

Aby wyznaczyć moment bezwładności opisanego danymi wejściowymi układu punktów materialnych względem zdefiniowanej prostej należy skorzystać z informacji, że moment bezwładności jest zawsze funkcją kwadratu odległości elementów masy od osi obrotu i dla pojedynczego punktu o masie *m* obracającego się w odległości *R* definiuje się go następująco:



$$I = mR^2$$

Gdzie m – masa, R – odległość punktu od prostej

W przypadku wielu punktów, ich łączny moment bezwładności jest sumą momentów poszczególnych punktów, definiuje się go następująco:

$$I = \sum_i m_i R_i^2$$

Gdzie m – masa, R – odległość punktu od prostej

STRUKTURY DANYCH

Aplikacja kliencka składa się z klasy EjbClient oraz interfejsu IDataMonitor. Poniżej przedstawione zostały metody klasy EjbClient:

```
void main(String[] args)
IDataMonitor getDataMonitor()
void printResult()
void processAxisParameters(IDataMonitor data)
void calculateMomentOfInertia(IDataMonitor data)
```

Metoda main jest punktem startowym aplikacji klienckiej. Wywołane zostają w niej metody getDataMonitor, processAxisParameters, calculateMomentOfInertia, printResult. Metoda getDataMonitor odpowiada za nawiązanie połączenia oraz zwrócenie obiektu typu IDataMonitor. Następnie następuje procesowanie parametrów osi obrotu, które odbywa się w metodzie processAxisParameters. Metoda calculateMomentOfInertia odpowiada za wyznaczenie momentu bezwładności. Ostatnim elementem jest metoda printResult, która odpowiada za wypisanie rezultatu.

URUCHOMIENIE

Proces kompilacji jest możliwy z użyciem komendy:

```
javac -cp <app-server-modules> -Xlint EjbClient.java IDataMonitor.java
```

Uruchomienie programu możliwe jest z użyciem komendy:

```
java -cp <app-server-modules> EjbClient
```