

Praca domowa 2 – cone

Przemysław Kleszcz

Zadanie

Cel pracy domowej: Wykorzystując metodę Monte Carlo, odnaleźć masę rzeczywistą analizowanego w zadaniu bloku materiału.

Organizacja strukturalna

Projekt zawiera jeden plik: **Cone.java**

Cone.java:

public class Cone

```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
protected void doPost(HttpServletRequest request, HttpServletResponse response)
private void fillConeSettings(HttpServletRequest request)
private Double getParameter(HttpServletRequest request, String parameter)
private void reset()
private double getConeMass()
private boolean createSpotAndCheck()
private double assumeMaxRay(double z)
private boolean checkSpot(double x, double y, double z, Defect defect)
```

class Defect

class ConeSettings

doGet(HttpServletRequest request, HttpServletResponse response) - Przyjmuje jako parametry obiekt żądania oraz obiekt odpowiedzi. Metoda przetwarzająca żądania GET servletu.

doPost(HttpServletRequest request, HttpServletResponse response) - Przyjmuje jako parametry obiekt żądania oraz obiekt odpowiedzi. Metoda przetwarzająca żądania POST servletu.

fillConeSettings(HttpServletRequest request) – Przyjmuje jako parametr obiekt żądania. Metoda wyodrębniająca z żądania informacje o współczynnikach badanej bryły.

getParameter(HttpServletRequest request, String parameter) – Przyjmuje jako parametry obiekt żądania oraz nazwę parametru. Metoda zwraca wartość parametru z obiektu żądania na podstawie jego nazwy.

reset() – Metoda nie przyjmuje parametrów. Zeruje zmienne zachowujące dane z wywołań servletu.

getConeMass() - Metoda nie przyjmuje parametrów. Zwraca masę rzeczywistą analizowanego w zadaniu bloku materiału.

createSpotAndCheck() - Metoda nie przyjmuje parametrów. Tworzy pojedynczy punkt dla metody Monte Carlo a następnie sprawdza czy współrzędne punktu znajdują się na obszarze defektu – jeśli tak, metoda zwraca wartość true, w przeciwnym wypadku false.

assumeMaxRay(double z) – Przyjmuje jako parametr wysokość z losowego punktu metody Monte Carlo. Zwraca promień przekroju poprzecznego stożka na przekazanej wysokości z.

checkSpot(double x, double y, double z, Defect defect) – Przyjmuje jako parametry współczynniki położenia losowego punktu metody Monte Carlo oraz obiekt defektu zawierający jego współczynniki. Zwraca true jeśli współrzędne punktu znajdują się w obszarze defektu, w przeciwnym wypadku false.

Defect – Klasa kontenerowa, zawierająca informacje o współczynnikach defektu.

ConeSettings – Klasa kontenerowa, zawierająca informacje o współczynnikach badanej bryły.

Opis mechanizmu

Program jest zrealizowany w konwencji servletu. Zadaniem metody przetwarzania żądań GET jest odczyt parametrów r, x, y, z defektu oraz dodanie ich do listy defektów, która będzie następnie wykorzystana w żądaniu POST. W metodzie przetwarzania żądań POST, na początku odczytujemy parametry ra, rb, h, c, g i zapisujemy w polu parametrów stożka. Obliczana jest objętość stożka na podstawie przesłanych danych. W następnej kolejności uruchamiana jest funkcjonalność wyznaczania masy stożka za pomocą metody Monte Carlo.

Funkcjonalność obliczania masy stożka zakłada wykonanie 1 miliona iteracji w czasie, których wyznaczony zostanie jeden losowy punkt dla każdego kroku. Współrzędne każdego z wygenerowanych punktów muszą znajdować się w granicach stożka, dlatego przy losowaniu pozycji punktów należy wyznaczyć promień przekroju poprzecznego dla wylosowanej wysokości z. W następnej kolejności punkt jest sprawdzany pod kątem zawierania się w obszarze defektów. Punkty zawierające i niezawierające się w obszarze defektów są osobno zliczane. Następnie:

- Obliczamy objętość obszaru defektów:

$$\text{objętość obszaru defektów} = \frac{\text{liczba punktów w obszarze defektów}}{\text{ilość wszystkich punktów}} * \text{objętość stożka}$$

- Obliczamy objętość obszaru bez defektów:

$$\text{objętość obszaru bez defektów} = \frac{\text{liczba punktów w obszarze bez defektów}}{\text{ilość wszystkich punktów}} * \text{objętość stożka}$$

- Ostatecznie wyznaczamy masę bloku materiału:

$$\text{masa bez defektów} = \text{objętość obszaru bez defektów} * \text{gęstość obszaru bez defektów}$$

$$\text{masa defektów} = \text{objętość obszaru defektów} * \text{gęstość obszaru defektów}$$

$$\text{masa bloku materiału} = \text{masa bez defektów} + \text{masa defektów}$$

Obliczona masa jest zaokrąglana do trzech miejsc po przecinku i umieszczana w odpowiedzi.
Na samym końcu uruchamiana jest funkcjonalność resetująca dane servletu.