Institut für Numerische und Angewandte Mathematik
FB Mathematik und Informatik der Universität Münster
Prof. Dr. Christian Engwer, MSc. Marcel Koch

02.11.2017

Übung zur Vorlesung

Wissenschaftliches Rechnen

WS 2017/18 — Blatt 4

Abgabe: 09.11.2017, 10:00 Uhr, Briefkasten 111
Code zusätzlich per e-mail an `marcel.koch@uni-muenster.de`

Aufgabe 1 (Barnes Hut für das n-Körper Problem) (8 Punkte)

In dieser Aufgabe sollen sie den Barnes-Hut Algorithmus aus der Vorlesung für das n-Körper Problem implementieren. Eine ausführlichere Beschreibung des Algorithmus finden Sie zum Beispiel unter `http://arborjs.org/docs/barnes-hut`.

(a) Gegeben seien die Punkte

Punkt	r	m
A	$(0.1, 0.8)$	1
B	$(0.95, 0.55)$	2
C	$(0.9, 0.95)$	3
D	$(0.8, 0.7)$	4
E	$(0.9, 0.4)$	5

Fügen Sie diese in der gegebenen Reihenfolge in einen Quadtree ein. Zeichnen Sie nach jedem eingefügten Knoten den Baum. Dabei soll für jeden bereits eingefügten Punkt erkenntlich sein, in welchem Knoten er ist. Des Weiteren soll für jeden Knoten die dort momentan gespeicherte Masse erkenntlich sein.

(b) Betrachten Sie ihren fertigen Baum aus (a). Markieren Sie die Knoten, die verwendet werden würden, um die Gesamtkraft auf den Körper A zu berechnen. Begründen Sie ihre Antwort durch geeignete Rechnungen. Wählen Sie dabei $\theta = 0.5$.

(c) Schreiben Sie ihr Programm von letzter Woche jetzt so um, dass zur Berechnung der Beschleunigung der Barnes-Hut Algorithmus verwendet wird. Nutzen Sie dafür das Framework von der Homepage. Das Framework ähnelt dem von letzter Woche in leicht modifizierter Form. Damit Sie sich auf Quadtrees beschränken können, geht das Framework von zweidimensionalen Orts- und Geschwindigkeitsvektoren (Datentyp Vector2D) aus. Des Weiteren sollte DataVariant1 verwendet werden. Der Datentyp `Body` hat eine zusätzliche Methode zum Vergleichen bekommen. Gegeben ist außerdem eine neue Datenstruktur `QuadTreeNode`:

```
struct QuadTreeNode
{
  QuadTreeNode(Quadrant q)
  : body(), quad(q), children()
  {}
  QuadTreeNode()
  : body(), quad(), children()
  {}

  bool isLeaf = true;
  bool isEmpty = true;
  Body body;
  Quadrant quad;
  std::vector<QuadTreeNode> children;
};
```

Sie können die Datenstruktur mit entsprechenden Funktionen erweitern, die den Zugriff auf die vier Kinderknoten erleichtern, wie z.B. `QuadTreeNode& childNO() { return children[0];}`

- Jeder Knoten vom Typ `QuadTreeNode` braucht ein Objekt des Datentyps `Quadrant`, das den zum Knoten gehörigen Quadranten speichert. Schreiben Sie diesen. Er soll die linkere untere und die rechte obere Ecke des Quadranten als Daten enthalten. Außerdem soll er folgende Methoden

  ```
  bool contains(const Vector2D& point)
  double length() const
  Quadrant NO()
  Quadrant SO()
  Quadrant SW()
  Quadrant NW()
  ```

  zur Verfügung stellen. Die ersten beiden sollen zurückgeben, ob ein Punkt im Quadranten enthalten ist sowie die Länge des Quadranten. Die restlichen sollen neue Quadranten erzeugen, die jeweils der NO, SO, SW beziehungsweise NW Ecke des ursprünglichen Quadranten entsprechen. Schreiben Sie zwei Konstruktoren, einen default Konstruktor der das Einheitsquadrat zuweist und einen Konstruktor der eine linke untere Ecke und eine rechte obere Ecke als Eingabe bekommt.

  ```
  Quadrant(const Vector2D ll, const Vector2D ur)
  Quadrant()
  ```

- Schreiben Sie Ihre Klasse zur Berechnung der Beschleunigung so um, dass es eine Methode

  ```
  void setupTree(const DataVariant1& data)
  ```

gibt, die nacheinander entsprechend dem Barnes-Hut Algorithnmus alle Body-Elemente aus `data` in einen Quadtree einfügt.

- Schreiben Sie die `evaluate` Methode der Beschleunigungsklasse so um, dass sie den Barnes-Hut Algorithmus zur Berechnung der Beschleunigung, die auf einen Körper wirkt, verwendet.

(d) Messen Sie im 'collision' Beispiel die Zeit für $N \in \{100, 200, ..., 1000\}$ mit dem Brute-Force Algorithmus von letzter Woche und dem Barnes-Hut Algorithmus von dieser Woche. Verwenden Sie $\Delta t = 1$, $o = 1000000$ und $T = 100$. Welche Abhängigkeit der Laufzeit des Programms von N ist zu erkennen?

Aufgabe 2 (n-Körper Problem aus Sicht des Schwerpunktsystems) (2 Punkte)

Wir betrachten erneut das n-Körper Problem mit den Notationen aus der Vorlesung. Der Schwerpunkt R der n Körper ist definiert durch

$$R := \frac{1}{M} \sum_{i=0}^{N-1} m_i r_i \quad \text{mit} \quad M := \sum_{i=0}^{N-1} m_i,$$

seine Geschwindigkeit V ergibt sich durch

$$V := \frac{1}{M} \sum_{i=0}^{N-1} m_i v_i.$$

Zeigen Sie, dass R sich mit konstanter Geschwindigkeit bewegt.

Aufgabe 3 (Hamilton-Operator als erstes Integral) (2 Punkte)

Zeigen Sie, dass der Hamilton-Operator $H(p, q)$ für Hamilton-Systeme, d.h. für Systeme mit

$$\dot{p} = \nabla_q H(p, q), \quad \dot{q} = -\nabla_p H(p, q),$$

ein erstes Integral ist.