

OPR Praktikum

Übung: Atomkern

Thomas Mahr

19. März 2018

1 Lernziele

- Operatoren überladen

2 Voraussetzungen

Kapitel *Operatoren* aus dem zweiten Vorlesungsteil *Vertiefung*.

3 Aufgabe: Atomkern

Ein Atomkern besteht aus Protonen und Neutronen. Das Stickstoffisotop ${}^{15}_7\text{N}$ besitzt 7 Protonen (Ordnungszahl=7) und 8 Neutronen (Massenzahl=7+8=15). Die Klasse `Atomkern` beschreibt Atomkerne.

Entwerfen und implementieren Sie die Klasse `Atomkern`. Diese Klasse muss folgende funktionalen Anforderungen erfüllen:

1. Ein Atomkern muss seine Ordnungszahl zurückliefern.
2. Ein Atomkern muss seine Massenzahl zurückliefern.
3. Ein Atomkern muss sein Elementsymbol (z.B. He für Helium) zurückliefern.
4. Das Elementsymbol muss veränderbar sein (denn bei Kernfusion und Kernspaltung ändert sich das Element).
5. Für die Verschmelzung zweier Atomkerne `k1` und `k2` müssen die Operatoren `+` und `+=` überladen werden: `k3 = k1 + k2` und `k1 += k2`
6. Für die Spaltung zweier Atomkerne `k1` und `k2` müssen die Operatoren `-` und `--` überladen werden: `k3 = k1 - k2` und `k1 -= k2`
7. Der Operator `*` muss so überladen werden, dass die Neutronen- und Protonenzahl mit einer skalaren Größe multipliziert werden kann: `k1 = 2 * k2`
8. Der Gleichheitsoperator `==` muss so überladen werden, dass er die Gleichheit zweier Atomkerne prüft. Zwei Atomkerne sind gleich, wenn sie dieselbe Ordnungszahl und dieselbe Massenzahl besitzen.
9. Ein Atomkern `k` muss über den Operator `<<` in einen Ausgabestrom ausgegeben werden, z.B.: `cout << k`

10. Stellen Sie statische, konstante Klassenelemente zur Repräsentation eines Protons und Neutrons zur Verfügung.

Für die Abnahme der Aufgabe gilt:

1. Zur Überprüfung der funktionalen Anforderungen wird eine Klasse `PrueffolgeAtomkern` zur Verfügung gestellt, die die vorgegebenen Prüfungen abarbeitet.
2. Sie dürfen die Dateien `main.cpp`, `Prueffolge.h`, `Prueffolge.cpp`, `PrueffolgeAtomkern.h` und `PrueffolgeAtomkern.cpp` nicht ändern.
3. Für die Abnahme der Aufgabe müssen alle Prüfungen fehlerfrei durchlaufen werden.

Hinweis: In der Musterlösung enthält die Klasse `Atomkern` 56 Codezeilen (ohne Kommentare und ohne Leerzeilen).

4 Aufgabe: Reaktor

Ein Reaktor dient der Spaltung und Verschmelzung von Atomkernen. Entwerfen und implementieren Sie die Klasse `Reaktor`. Diese Klasse muss folgende funktionalen Anforderungen erfüllen:

1. Mittels des Operators `<` wird dem Reaktor `r` ein Atomkern `k` hinzugefügt: `r < a;`
2. Mittels des Operators `>` wird dem Reaktor `r` ein Atomkern `k` entnommen: `r > a;`
3. Der Operator `*` liefert den Inhalt des Reaktors zurück: `Atomkern k = *r;`
4. Der Operator `~` entleert den Reaktor: `~r;`

Abnahme der Anforderungen:

1. Zur Überprüfung der funktionalen Anforderungen wird eine Klasse `PrueffolgeReaktor` zur Verfügung gestellt, die die vorgegebenen Prüfungen abarbeitet.
2. Sie dürfen die Dateien `main.cpp`, `Prueffolge.h`, `Prueffolge.cpp`, `PrueffolgeReaktor.h` und `PrueffolgeReaktor.cpp` nicht ändern.
3. Für die Abnahme der Aufgabe müssen alle Prüfungen fehlerfrei durchlaufen werden.

Hinweis: In der Musterlösung enthält die Klasse `Reaktor` 40 Codezeilen (ohne Kommentare und ohne Leerzeilen).