

iso21

Ein Projekt im Rahmen der Vorlesung

Praxis des Programmierens

Martin-Luther-Universität Halle-Wittenberg

Inhalt

1. Die Aufgabe	3
2. Die Lösung	4
a. Die Eingabe	4
b. Die Isotopenberechnung.....	5
c. Die Ausgabe	6
d. Die grafische Oberfläche.....	8
3. Das Nachwort.....	10
4. Der Dank	10

1. Die Aufgabe

Die Aufgabe erteilte uns Prof. Dr. Csuk (Leiter des Instituts für Chemie an der Martin-Luther-Universität Halle-Wittenberg).

Isotopenverteilung einer chemischen Verbindung

Ziel ist die Erarbeitung eines Programms zur Berechnung des Isotopenverteilungsmusters unter Berücksichtigung der natürlichen Isotopenverteilung. Die Eingabe erfolgt als Formel (Bsp. CH₂ oder (CH₃)₂CHCH(CH₃)₂). Das Programm soll die Elemente des Periodensystems (Hf -> Hafnium, HF -> Fluorwasserstoff) sowie das Element „D“ für Deuterium erkennen.

Als Ausgabe wurde einerseits eine Tabelle, welche über das Clipboard kopierbar ist, andererseits eine speicherbare Grafik der berechneten Isotope gewünscht.

Optional soll ein „Labelling“ (in der Grafik werden die berechneten Werte angezeigt) und eine „Hohe Auflösung“ (Die Werte werden auf 6 Stellen genau berechnet) einschaltbar sein.

Die Farben der Grafik müssen so gewählt sein, dass ein Schwarz-Weiß-Ausdruck zu sinnvollen Ergebnissen führt.

2. Die Lösung

Wir entschieden uns, das Projekt in der Sprache „Java“ zu programmieren, da dies eine für uns noch neue Programmiersprache war und Übung bekanntlich den Meister macht.

Schon nach den ersten beiden Tagen hatten wir uns Gedanken über die interne Speicherung gemacht und einen Ansatz gefunden, den es auszuarbeiten galt.

Alle Elemente des Periodensystems, welche stabile Isotope besitzen, müssen mit ihren Isotopen gespeichert werden, um die Eingabe zu überprüfen und eine korrekte Berechnung gewährleisten zu können. So erstellten wir eine Klasse *PerSys*, welche ein *Array* vom Typ *Element* enthält. Jedes *Element* speichert für sich seinen Namen, sein chemisch eindeutiges Zeichen und eine Schlange vom Typ *isos*, in die die jeweiligen Isotope mit Gewicht und prozentualem Vorkommen gespeichert sind. (Abb. 2.1)

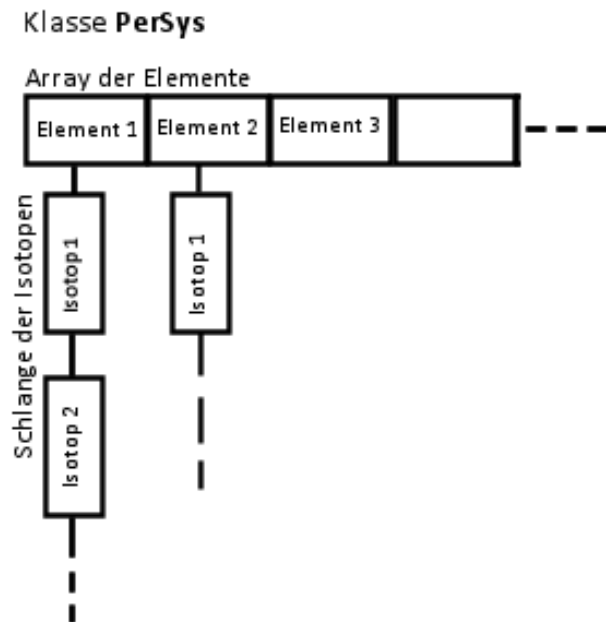


Abb 2.1 Die Interne Speicherung der Elemente mit ihren jeweiligen Isotopen

Das Programm soll nun folgende Funktionen ermöglichen:

- In ein Textfeld kann eine Chemische Verbindung eingegeben werden.
- Durch einen Klick auf einen Knopf werden die Isotope berechnet.
- Nach der Berechnung werden die Ergebnisse tabellarisch und grafisch ausgegeben.

2. a Die Eingabe

Die Eingabe muss „case sensitive“, also mit Beachtung der Groß- und Kleinschreibung, erfolgen, da zum Beispiel Hf (Hafnium) eine andere Verbindung darstellt als HF (Flourwasserstoff).

Es lassen sich folgende Regeln aufstellen:

- Ein neues Element fängt mit einem Großbuchstaben an.
- Einem Großbuchstabe folgt optional ein Kleinbuchstaben, welcher das Elementzeichen vervollständigt, oder/und eine Zahl, welche die Anzahl des vorkommenden Elements angibt.

Also haben wir eine Eingabeklasse geschrieben.

Der Eingabestring wird analysiert. Das erste Zeichen muss ein Großbuchstabe sein. Dieser wird in einen neuen temporären *String* geschrieben. Ist das nächste Zeichen ein Kleinbuchstabe, wird dieser an den *String* gehängt. Wenn nun ein Großbuchstabe folgt, ist das Element vollständig und wird in einen *Vektor* vom Typ *element_object* gepackt und bekommt den Wert 1 für die Anzahl. Folgt jedoch eine Zahl, wird das Element in den Vektor mit dem Wert der Zahl eingefügt.

Dem Anwender ist es möglich Klammern in seinem Ausdruck zu setzen, was die Eingabe deutlich erleichtert. Die Überprüfung ist durch einen Algorithmus implementiert, der auf der Datenstruktur „*Stack*“ basiert. Alle Klammern werden nacheinander auf einen Stapel gelegt. Folgt auf eine offene Klammer eine geschlossene werden beide aus dem „*Stack*“ entfernt. Wenn dieser am Ende der Prüfung leer ist, so ist der Ausdruck syntaktisch korrekt geklammert. Durch rekursive Auswertung des Klammerinhalts, ist es sogar möglich verschachtelte Klammerausdrücke auszuwerten.

Beim Einfügen in den Vektor wird jedesmal überprüft, ob das Element in unserem Periodensystem (*PerSys*) existiert und, ob das Element bereits in dem Vektor gespeichert wurde, ist dies der Fall, wird die aktuelle Anzahl des Elements einfach auf die Anzahl, des im Vektor vorhandenen Elements, addiert.

Somit erhalten wir einen Vektor mit allen nötigen Elementen und können damit die Berechnung starten.

2. b Die Isotopenberechnung

Um die Isotope zu speichern benötigen wir eine weitere Datenstruktur.

Schon nach kurzem Überlegen war klar, *Arrays* sind genauso unbrauchbar, wie *Schlangen*, *Stacks* oder *Listen*, denn schon bei Standardverbindungen wie der Glucose ($C_6H_{12}O_6$) wird deutlich mit wie vielen Isotopen wir arbeiten müssen:

- Kohlenstoff hat 2 Isotope und kommt 6 mal vor: $2^6 = 64$
- Wasserstoff hat 2 Isotope und kommt 12 mal vor: $2^{12} = 4096$
- Sauerstoff hat 3 Isotope und kommt 6 mal vor: $3^6 = 729$

Da jede mögliche Verbindung vorkommen kann, erhält man also $64 * 4096 * 729 = 191.102.976$ Isotope, die jedes *Array* sprengen würden.

Also brauchen wir eine intelligendere Struktur, die nur Isotope speichert, die von Interesse sind (Prozent ≥ 0.01).

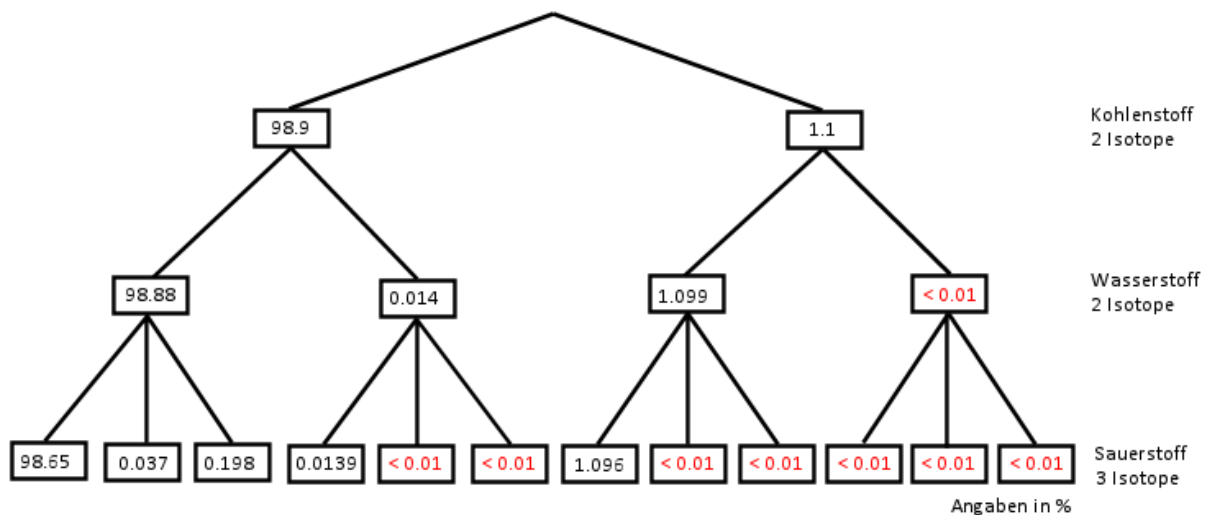


Abb. 2.2: Isotopenberechnung am Beispiel CHO. Die Knoten mit roten Werten werden nicht angelegt.

Wir entschieden uns für einen *Baum*, der in den *Knoten* einer Ebene die Isotope eines Elementes speichert und schon die aktuelle Prozentzahl enthält. Sollte ein *Knoten* ein *Kind* bekommen, welches weniger als 0.01 % aufweist, wird dieses nicht angelegt. (Abbildung 2.2 verdeutlicht das Prinzip für die Verbindung CHO)

Ist das Programm beim letzten Element angelangt (die untersten Blätter des Baumes, im Beispiel die schwarzen Werte des Sauerstoffs), werden alle Elemente in einen Vektor geschrieben und die Berechnung ist erfolgreich beendet.

2. c Die Ausgabe

Die Ausgabe soll in tabellarischer und grafischer Form geschehen.

Wir haben jetzt einen Vektor, in dem alle Isotope der eingegebenen Verbindung gespeichert sind.

Für die tabellarische Ausgabe haben wir eine *Textarea* eingerichtet, in die wir mit einer einfachen Routine alle Isotope aus dem Vektor schreiben. Der Inhalt kann per Knopfdruck in den Zwischenspeicher kopiert und in jeden anderen Editor eingefügt werden.

Die grafische Ausgabe erforderte jedoch etwas mehr Überlegung:

Die Idee ist ein Achsenkreuz. Auf der vertikalen Achse werden die Prozente eingetragen und auf der horizontalen die Gewichte.

Wir bestimmen das größte und das kleinste Gewicht und können so die horizontale Achse einteilen.

Jetzt geht eine Routine durch den Vektor und sucht sich den richtigen Punkt auf der horizontalen Achse. Nun wird ein Balken gezeichnet vom Startpunkt auf der Achse senkrecht nach oben, relativ zum Prozentwert, wobei ein Prozent zwei Pixeln entspricht.

Die erstellte Grafik kann über einen Klick auf einen Knopf als .jpg-Datei gespeichert werden. (Abb. 2.3)

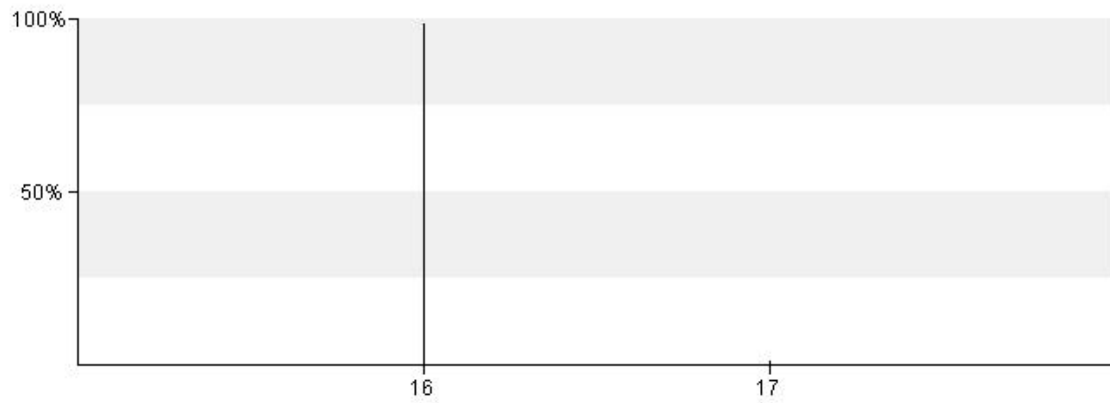
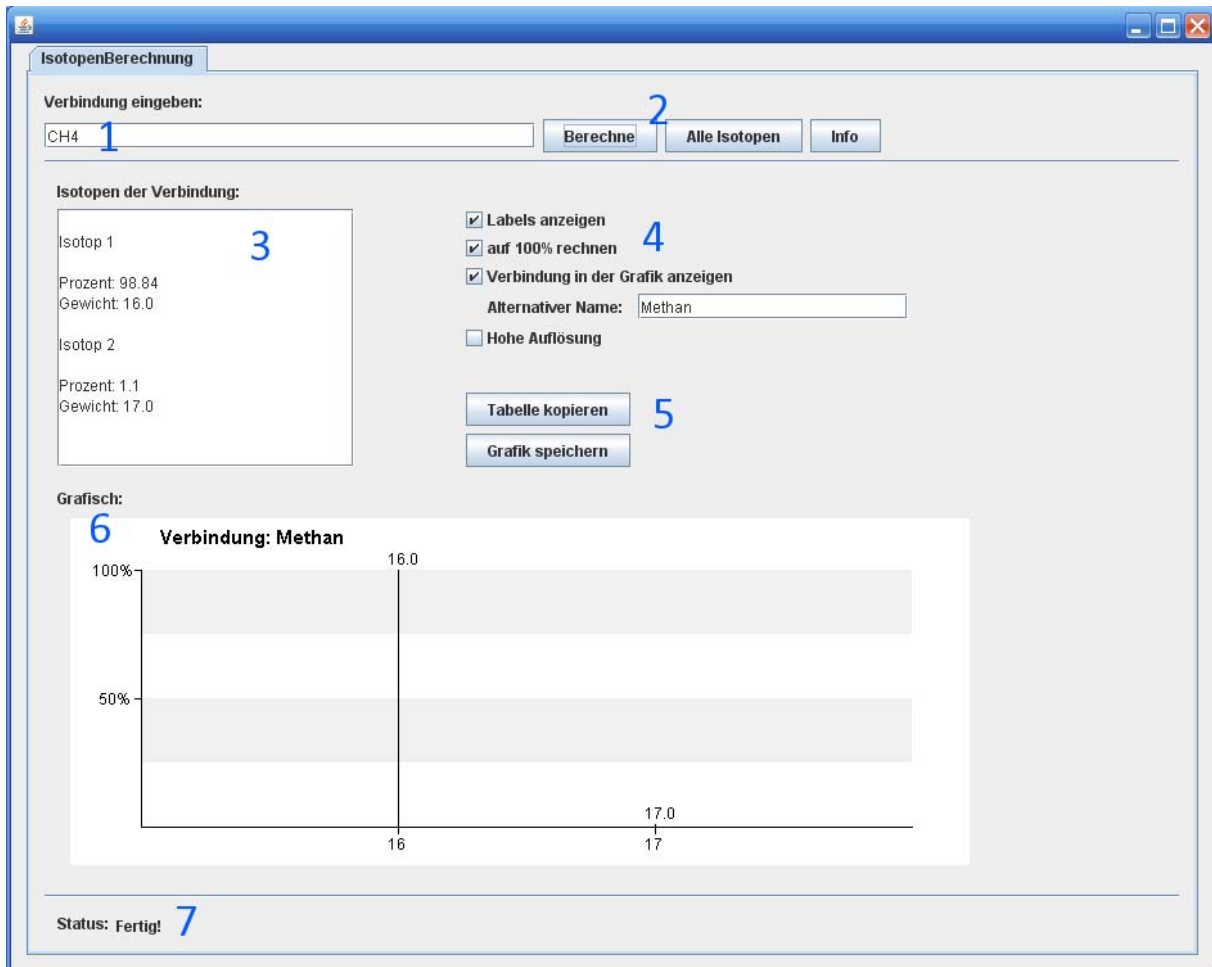


Abb 2.3 Eine mit diesem Tool erstellte Grafik der Verbindung CH4

2. d Die grafische Oberfläche



IsotopenBerechnung

Verbindung eingeben:

CH4 1

Berechne 2 Alle Isotopen Info

Isotopen der Verbindung:

Isotop 1 3	<input checked="" type="checkbox"/> Labels anzeigen 4
Prozent: 98.84	<input checked="" type="checkbox"/> auf 100% rechnen
Gewicht: 16.0	<input checked="" type="checkbox"/> Verbindung in der Grafik anzeigen
Isotop 2	Alternativer Name: Methan
Prozent: 1.1	<input type="checkbox"/> Hohe Auflösung
Gewicht: 17.0	Tabelle kopieren 5
	Grafik speichern

Grafisch: 6

Verbindung: Methan

100% 16.0

50%

17.0

16 17

Status: Fertig! 7

1 - Das Eingabefeld:

- Hier wird die Verbindung eingegeben

2 - Die 3 Buttons

- Berechnen: Mit einem Klick werden die Isotope der eingegebenen Verbindung berechnet und im Ausgabefeld (3) sowie grafisch (6) angezeigt
- Alle Isotope: Mit einem Klick werden alle existierenden Isotope im Ausgabefeld (3) aufgelistet
- Info: Mit einem Klick wird eine Infografik (6) angezeigt

3 - Das Ausgabefeld

- Hier werden die berechneten Isotope der Verbindung aufgelistet

4 - Die optionalen Checkboxes

- Labels anzeigen: Beschriftet die Balken mit den berechneten Gewichten
- Auf 100% rechnen: Nimmt den höchsten errechneten Prozentwert als 100% und gleicht alle anderen Werte relativ dazu an (hat nur Auswirkungen auf die Grafik (6))
- Verbindung anzeigen: Es wird die Verbindung oben links in der Grafik (6) angezeigt, optional kann ein alternativer Name im Textfeld angegeben werden
- Hohe Auflösung: Die Gewichte werden auf 6 Stellen nach dem Komma ausgegeben (ist dies nicht aktiviert, werden die Gewichte auf ganze Zahlen gerundet)

5 - Die 2 Buttons

- Tabelle kopieren: Die im Ausgabefeld (3) stehenden Isotope werden in den Zwischenspeicher kopiert
- Grafik speichern: Speichert die Grafik als .jpg-Datei

6 - Die Grafik

- In diesem Bereich wird die Grafik angezeigt

7 - Der Status

- Hier wird der aktuelle Status des Tools bekannt gegeben.

3. Das Nachwort

Die in dieser Dokumentation vorgestellten Beispiele und Lösungen sind natürlich nur ein kleiner Teil des Programms und wurden nur ansatzweise erklärt. Bitte habt Verständnis, dass wir hier nicht auf alle Aspekte eingehen konnten und dieses Script nicht den gesamten internen Hintergrund des Programms zeigt.

4. Der Dank

Unser Dank gebührt natürlich Herrn Prof. Dr. Csuk für das Vertrauen, das er in uns gesteckt hat, und Herrn Dr. Blaar, der uns bei der Bearbeitung stets unter die Arme griff.

Außerdem wollen wir den Mitarbeitern der „Cafeteria Ein-Stein“ danken, die uns nach den vielen Nachtschichten mit ihrem Kaffee wieder zurück in die Realität geholt haben!