

anwendung Linearer Programmierung

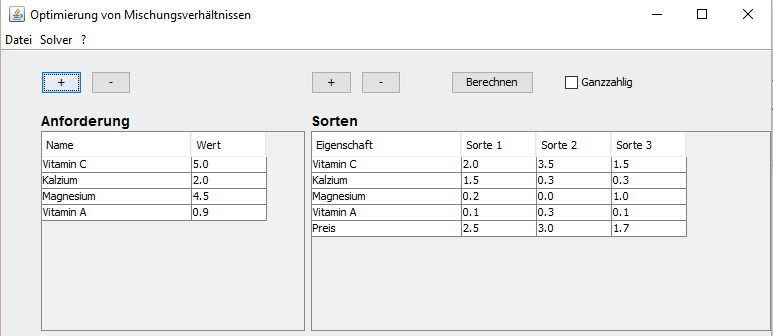
Mischungsplanung 3.0



23. Juni 2016

Marius Germann 288990

BENEDIKT GOOS 289839



Inhaltsverzeichnis

[1. Einleitung: 3](#_Toc453160822)

[2. Durchgeführte Verbesserungen: 3](#_Toc453160823)

[a. Programmberechnung wiederherstellen: 3](#_Toc453160824)

[b. Funktion Neu: 3](#_Toc453160825)

[c. Funktion Speichern: 4](#_Toc453160826)

[d. Funktion Öffnen: 4](#_Toc453160827)

[e. Funktion Beenden: 4](#_Toc453160828)

[f. Hilfeaufruf: 4](#_Toc453160829)

[g. Benutzerfreundlichkeit 5](#_Toc453160830)

[h. Hilfeseite 5](#_Toc453160831)

[3. Testläufe: 6](#_Toc453160832)

[4. Schlusswort 7](#_Toc453160833)

# Einleitung:

Aus dem Pflichtenheft ist ersichtlich, welchen Zustand die Methode „Mischungsplaner“ vor der Optimierung hatte. Unsere Aufgabe war es nun, die im Pflichtenheft definierten Fehler ausfindig zu machen und zu beheben. Dieser Bericht soll die ausgeführten Schritte zur Verbesserung des Mischungsplaners dokumentieren. Ebenfalls soll durch den Abschnitt „Test“ bewiesen werden das der Solver nun wieder einsetzbar ist und optimiert arbeitet.

# Durchgeführte Verbesserungen:

Damit der Mischungsplaner die im Pflichtenheft definierten Aufgaben erfüllen kann waren verschiedene Verbesserungsmaßnahmen nötig. Diese werden hier der Reihe nach aufgezeigt und dokumentiert. Leider ist uns nicht bekannt welcher Code im Mischungsplanung2013.jar verwendet wurde. Im Vergleich zum vorhandenen Source-Code muss dieser aber verschieden sein, da im Source-Code verschiedene Funktionen bereits funktionsfähig waren.

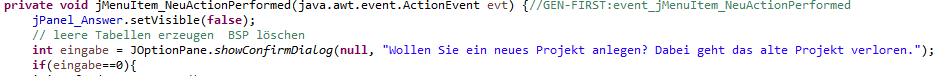
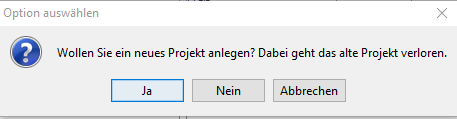
# Programmberechnung wiederherstellen:

Das Programm war nicht berechnungsfähig. Daher mussten wir als erstes die Berechnungsfähigkeit wiederherstellen. Dazu haben wir den vorhanden Source-Code in ein Eclipse Projekt angelegt. Danach musste der Pfad so angepasst werden, dass im angegebenen Ordner die Datei «LP\_SOLVE.EXE» vorhanden ist. Mit dem aktuellen Source-Code erfolgt die Berechnung wieder fehlerfrei.



# Funktion Neu:

Im alten Programm wurde bei einem Klick auf „Neu“ sofort ein leeres Tabellenblatt angelegt. Unsere Aufgabe war es, hier noch eine Abfrage einzufügen.   
Dazu haben wir ein OptionPane mit dem Dialog: «Wollen Sie ein neues Projekt anlegen? Dabei geht das alte Projekt verloren». implementiert. Der Nutzer hat anschliessend die Möglichkeit «Ja», «Nein» oder «Abbrechen» zu klicken. Im Code sieht diese Funktionalität folgendermassen aus:



# Funktion Speichern:

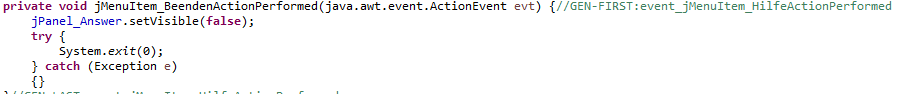
Beim Mischungsplaner2013.jar funktionierte das Speichern nicht. Sobald wir das Programm mit dem vorhandenen Source-Code in Eclipse integriert hatten, funktionierte diese Funktion ebenfalls.

# Funktion Öffnen:

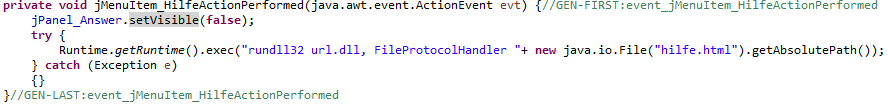
Im alten Programm war das Öffnen fehleranfällig. Beim Öffnen eines angelegten Projekts wurde nur die erste Tabelle mit den Anforderungen geladen. Die Tabelle mit «Zur Auswahl stehende Sorten» wurde nicht geladen.  
Im neu Integrierten Programm in Eclipse war diese Funktion ebenfalls schon funktionsfähig.

# Funktion Beenden:

Im alten Programm hatte der Button «Beenden» keine Funktion. Diesem haben wir nun die Funktion „Beenden“ zugewiesen, so dass das Programm beendet wird. Dazu haben wir folgenden Java-Code eingebaut:

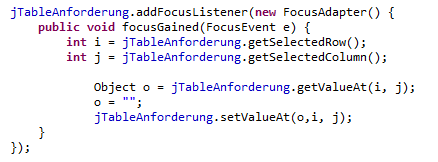


# Hilfeaufruf:

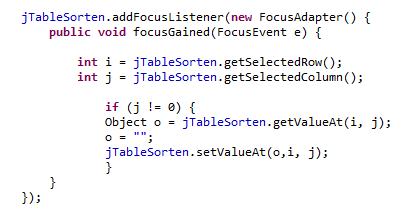
Der Hilfe-Button war ebenfalls ohne Funktion. Diese wurden geändert. Ebenfalls haben wir eine übersichtlichere Hilfe implementiert. Voraussetzung für eine funktionierende Hilfefunktion ist, dass die Datei hilfe.html im gleichen Ordner wie die ausgeführte .jar Datei ist.

# Benutzerfreundlichkeit

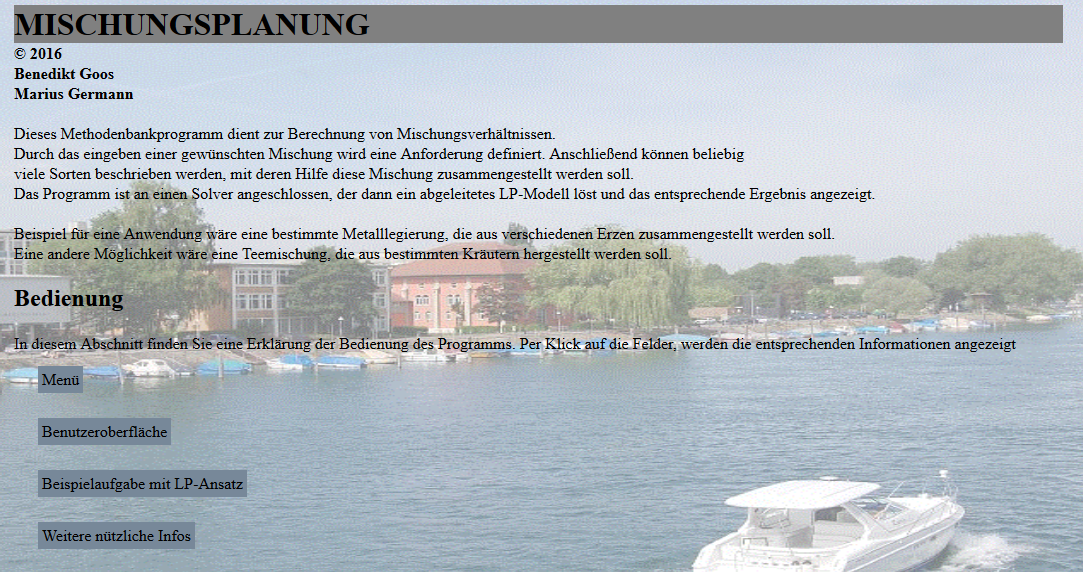
Um die Benutzerfreundlichkeit zu erhöhen wurde ein FocusListener implementiert, der sobald in der Tabelle „Anforderungen“ in das Feld „Wert“, mit dem vordefinierten Wert „0.0“ geklickt wird, diesen entfernt.



Dasselbe wurde für die Tabelle „Sorten“ vorgenommen. Jedoch mit dem Unterschied, dass die if-Anweisung verhindert, dass die übernommenen Werte aus der Anforderung automatisch gelöscht werden, sobald der Fokus auf ihnen liegt.



# Hilfeseite

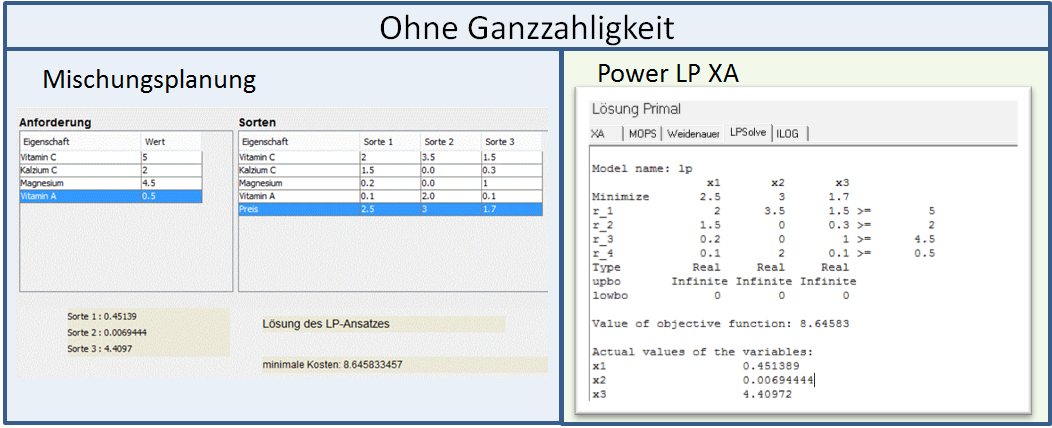
Eine Hilfeseite wurde für den Mischungsplaner implementiert, welche eine Beschreibung der Methode und eine Bedienungsanleitung enthält. Die Bedienungsanleitung verfügt über eine Menüleiste, die beim Klicken auf die Menüpunkte den der dazugehörige Text öffnet.

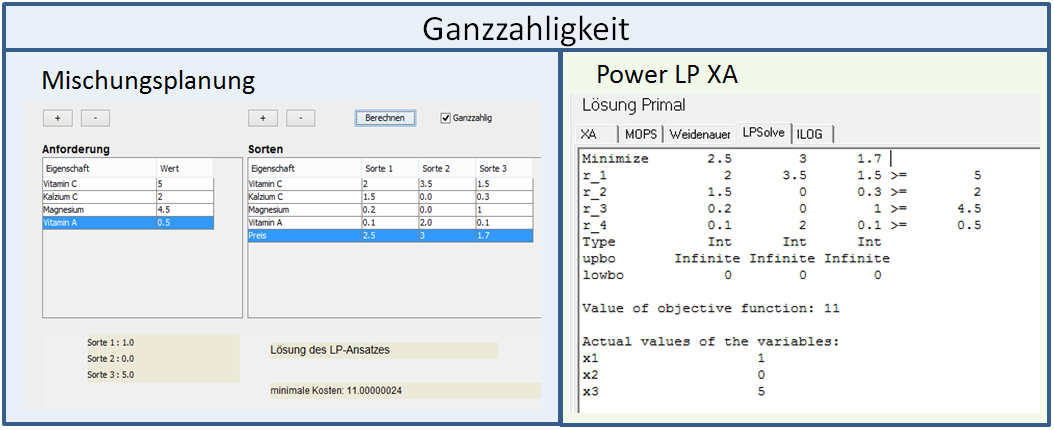
# Testläufe:

Um zu beweisen, dass der Mischungsplaner korrekt rechnet, wurde die folgende Beispielaufgabe sowohl mit dem Mischungsplaner als auch mit dem Power LP – XA gelöst und die Ergebnisse miteinander verglichen.

**Anforderungen: Sorten:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Wert** |  | **Sorte 1** | **Sorte 2** | **Sorte 3** |
| Vitamin C (in mg) | 5.0 |  | 2.0 | 3.5 | 1.5 |
| Kalzium (in mg) | 2.0 |  | 1.5 | 0.0 | 0.3 |
| Magnesium (in mg) | 4.5 |  | 0.2 | 0.0 | 1.0 |
| Vitamin A (in mg) | 0.5 |  | 0.1 | 2.0 | 0.1 |
| **Preis** |  |  | 2.5 | 3.0 | 1.7 |





Wie anhand der Screenshots zu sehen ist, kommen beide Solver auf das gleiche Ergebnis. Somit ist gezeigt, dass der Mischungsplaner und der Power LP – XA das LP-Modell identisch berechnen.

# Schlusswort

Alle definierten Verbesserungen wurden von uns durchgeführt. Nun steht ein Funktionsfähiger Solver zur Verfügung. Damit der Solver auch in Zukunft genutzt werden kann und die Änderungen Nachvollzogen werden können, liefern wir folgende Daten mit:

* Programm als ausführbare .jar Datei, über diese kann die Methode im OR\_Alpha eingebunden werden.
* Programm als Source-Code, falls zu einem späteren Zeitpunkt darauf zugegriffen werden muss.
* Benutzerhandbuch als PDF, soll eine Beschreibung über die Methode liefern.
* Hilfe.html, die neu entwickelte Hilfeseite. Diese muss immer im .jar Verzeichnis vorhanden sein, damit sie aufrufbar ist.
* Analysebericht