

Prüfungsplaner 0.9.10.5

Inhaltsverzeichnis

1. Einleitung	2
2. Installation	3
2.1. GUI vs. Kommandozeile	3
2.2. Installation	3
3. Ein- / Ausgabedateien	3
3.1. Modelle	3
3.1.1. interne / externe Prüfungen	4
3.2. Einstellungen	4
3.3. Planung	4
3.3.1. Planpool	4
3.3.2. manuelle Pläne	5
3.4. Überblick unterstützte Dateiformate	5
3.5. Modell-Verzeichnisse	5
3.5.1. Demo-Version	6
3.6. Zustandsdatei	6
3.7. Exportierte Pläne	6
3.8. Intendierte Nutzung	6
4. Nutzung der grafischen Oberfläche	7
4.1. Laden eines Modells.	7
4.2. Übersicht Reiter	8
4.3. Modell / Einstellungen bearbeiten	9
4.4. Planung durchführen	9
4.4.1. Erste Planungsphase: zufällige Generierung von Plänen	9
4.4.2. Zweite Planungsphase: Mutationen	10
4.4.3. Einstellungen für Planung	10
4.4.4. Threads	11
4.4.5. Steuerung	11
4.4.6. Export von Plänen	12
4.5. Manuelle Planung	12
4.5.1. Liste der manuellen Pläne	13
4.5.2. Manuelle Bearbeitung	13
4.5.3. Bewertungen	13
4.5.4. Automatische Optimieren / Einplanen	13
5. Nutzung Kommandozeilen-Tool	14
6. Bewertung von Plänen	14
6.1. Warnung vor unvorsichtigen Einstellungsänderungen	15
6.1.1. Beispiel	15
6.1.2. Negativbeispiel	15
6.1.3. Generelle Empfehlungen	16
6.2. Implizite Konflikte	16
6.3. Ressourcenkonflikte	16
6.3.1. Mehrfache Raumbelugung	16

6.3.2. Lehry mehrfach eingeplant	16
6.3.3. Prüfy mehrfach eingeplant	16
6.3.4. Schüly hat mehrere Prüfungen an gleichem Tag	17
6.4. Lokale Einzelbewertungen	17
6.4.1. Empfehlungen	17
6.5. Lehry Einzelbewertungen	18
6.5.1. Anzahl Prüfungstage	18
6.5.2. Viele Prüfungen pro Tag	18
6.5.3. Viele Prüfungen	18
6.5.4. Lange Pausen	18
6.5.5. Viele Prüfungen ohne Pause	19
6.5.6. Lehry / Prüfy - Raumwechsel ohne Pause	19
6.5.7. Lehry Raumwechsel mit Pause	19
6.6. Prüfungsgruppe Einzelbewertungen	19
6.6.1. Prüfung außerhalb Gruppenzeit	19
6.6.2. Mehrere Vorsitzendys / Protokollantys	19
6.7. Raum Einzelbewertungen	19
6.7.1. Anzahl Prüfungstage	19
6.7.2. Viele Prüfungen ohne Pause	20
6.7.3. Lehry- / Prüfy-Wechsel ohne Pause	20

1. Einleitung

Prüfungsplaner dient dazu, mündliche Abiturprüfungen zu planen. Es wurde für hessische Gymnasien entwickelt, sollte aber auch in vergleichbaren Situationen nützlich sein.

Eingabe des Programms ist eine Liste von Prüfungen, die Schülys bei bestimmten Lehrlys in einem Fach ablegen müssen (diese Dokumentation ist, wie auch Prüfungsplaner selbst, nach Phettberg entgendert). Das Programm soll für jede Prüfung jeweils

- ein Vorsitzendy und ein Protokollanty (zwei Lehrlys, die das Fach selbst unterrichten)
- ein Zeitfenster
- einen Raum

finden, so dass insgesamt ein *guter* Plan entsteht. Genau zu definieren, was ein guter Plan ist, ist kompliziert; eine genaue Erklärung bildet einen der Hauptteile dieses Dokumentes.

Es gibt offensichtliche Einschränkungen, wie dass ein Lehry nicht an 2 Prüfungen gleichzeitig teilnehmen kann, oder dass ein Raum nicht doppelt belegt sein darf. Für Fachfremdys weniger offensichtlich ist, dass ein Schüly maximal eine Prüfung pro Tag haben darf. Zudem werden Prüfungen zu Gruppen zusammengefasst. Prüfungen einer Gruppe müssen zeitlich sehr eng aneinander geplant werden. Dies erlaubt es, dass in einer Prüfungsgruppe das gleiche Thema geprüft wird. Schülys können sich aufgrund von Beaufsichtigung vor und nach den eigentlichen Prüfungen nicht über die Inhalte der Prüfungen abstimmen.

Neben den Prüfungen selbst werden weitere Eingaben benötigt:

- eine Liste aller für ein Fach zur Verfügung stehenden Lehrlys
- eine Liste aller Zeitfenster; Zeitfenster dürfen nicht überlappen
- eine Liste aller Räume, die für ein Fach geeignet sind
- eine Zuordnung von Prüfungen zu Gruppen
- weitere Einschränkungen und Vorlieben (z. B. Lehry kann an bestimmtem Tag nicht prüfen)

Ausgabe des Programms sind mehrere, hoffentlich gute Prüfungspläne. Prüfungsplaner erlaubt es, diese Pläne zusammen mit Ihrer Bewertung genau zu inspizieren und auch manuell zu bearbeiten.

Die Prüfungspläne werden zufällig erzeugt. Jeder Aufruf des Programms liefert andere Pläne. Es wird nicht garantiert, dass die bestmöglichen Pläne gefunden werden. Ziel ist vielmehr, relativ schnell gute Pläne zu finden.

Die Implementierung basiert auf einem genetischen Algorithmus. Idee ist, zunächst zufällig Pläne zu erzeugen. Dabei ist die Erzeugung nicht rein zufällig, sondern Wahrscheinlichkeiten werden so gewählt, dass die Entstehung guter Pläne wahrscheinlicher ist. Zum Beispiel wird für eine Prüfung ein Raum viel

eher gewählt, wenn der Raum noch nicht belegt ist. Es werden nur die besten so erzeugten Pläne behalten. Schlechte Pläne werden entsorgt. Danach werden kleine zufällige Änderungen (sogenannte Mutationen) durchgeführt und immer nur die besten resultierenden Pläne behalten. Eine naive Implementierung dieses Ansatzes würde dazu führen, schnell in Plänen festzustecken, die lokal optimal sind, also über einzelne Mutationen nicht verbessert werden können, insgesamt gesehen aber dennoch relativ schlecht sind. Es werden verschiedene Maßnahmen getroffen, um dies zu verhindern. Dies resultiert in der relativ schnellen Erzeugung guter Pläne. Diese sind aber im Allgemeinen nicht optimal und mehrfache Aufrufe des Programms resultieren in unterschiedlichen Plänen.

2. Installation

2.1. GUI vs. Kommandozeile

Prüfungsplaner gibt es als Kommandozeilen-Tool sowie mit grafischer Oberfläche. Beide Versionen verarbeiten die gleichen Eingabedateien auf die gleiche Weise und erzeugen die gleichen Ausgaben. Wie gewöhnlich ist die Kommandozeilenversion besser automatisierbar, benötigt weniger Ressourcen und kann einfacher auf verschiedene Systeme portiert werden. Allerdings erfordert sie ein gewisses Maß an Einarbeitung und Vertrautheit. (Kennt man die Kommandozeilenparameter, ist es schneller und einfacher diese zu nutzen, als sich durch Menüs zu klicken. Man muss die Parameter dafür aber kennen.) Die grafische Oberfläche für erfahrene Nutzer umständlicher und langsamer nutzbar, aber intuitiver. Zudem ist es über die grafische Oberfläche wesentlich einfacher, Pläne manuell zu bearbeiten.

Da die meisten Nutzers Prüfungsplaner wohl nur selten (einmal pro Jahr vor Prüfungen) verwenden, bietet sich wahrscheinlich meist die Nutzung der grafischen Oberfläche an. Soll Prüfungsplaner dagegen (teil)automatisiert oder z.B. über eine Netzwerkverbindung auf einem anderen Rechner genutzt werden, ist die Kommandozeilenversion vorteilhaft. Auch ein kombinierter Einsatz kann sinnvoll sein. So ist z.B. das lokale Bearbeiten der Eingabedateien mittels grafischer Oberfläche auf einem relativ langsamen Rechner und danach die eigentliche Suche nach guten Plänen mittels Kommandozeilen-Tool auf einem schnellen Rechner im Netzwerk denkbar.

2.2. Installation

Für Windows steht ein Installprogramm zur Verfügung. Dieser ist wie üblich zu nutzen und installiert je nach Auswahl die grafische Oberfläche, das Kommandozeilen-Tool, Dokumentation sowie Beispiel-Eingaben. Ebenso werden Start-Menüeinträge sowie ein Uninstaller-Eintrag angelegt. Sie finden Prüfungsplaner nach der Installation im Startmenü-Ordner *Prüfungsplaner*. Dort erhalten Sie auch Zugriff auf die Dokumentation sowie den Uninstaller.

Für Linux werden Binaries zur Verfügung gestellt.

Sollten Sie Probleme bei der Installation haben, die Sie auch nicht mit Hilfe von erfahrenen Computernutzern lösen können, kontaktieren Sie bitte den Autor Thomas Türk (kontakt@thomas-tuerk.de).

3. Ein- / Ausgabedateien

Prüfungsplaner benutzt *Modelle*, *Einstellungen* und die erstellte *Planung*. Diese können über bestimmte Dateien ein- und ausgegeben werden.

3.1. Modelle

Modelle enthalten die komplette Beschreibung der Planungsaufgabe, d.h.

- zu planenden Prüfungen
- Lehrers
- Schülers
- Räume
- Zeiten
- Einschränkungen und Vorlieben für die Planung
- ...

3.1.1. interne / externe Prüfungen

Prüfungen werden in *interne* und *externe* Prüfungen unterteilt. Bei externen Prüfungen handelt es sich um Prüfungen, die an einem nicht verwalteten Tag oder in einem nicht verwalteten Raum stattfinden. Alle anderen Prüfungen sind intern. Nur interne Prüfungen werden von Prüfungsplaner geplant und für die Bewertung berücksichtigt. Externe Prüfungen müssen bereits in der Eingabe vollständig geplant sein.

Da externe Prüfungen weder von Prüfungsplaner geplant, noch bewertet werden, könnten ohne große Probleme komplett aus den Eingaben entfernt werden. Hauptgrund für die Unterstützung externer Prüfungen ist, dass sie es erlauben aus LUST exportierte Prüfungslisten nicht nachbearbeiten zu müssen. Oft sind externe Prüfungen an einem nicht für die Planung verwendeten Tag. In diesem Fall sind sie wirklich nur wegen den einfacheren Exportes nützlich. Externe Prüfungen an einem verwalteten Tag müssen zu einer verwalteten Uhrzeit stattfinden. Prüfungsplaner benutzt dann die Information, dass die an einer solchen externen Prüfung beteiligten Lehrys und das Schüly nicht gleichzeitig für andere Prüfungen zur Verfügung stehen. Dies wäre jedoch genauso über Vorlieben möglich.

Ähnlich zu externen Prüfungen sind nicht vollständig geplante interne Prüfungen. Auch diese werden von Prüfungsplaner nicht geplant und dienen hauptsächlich dazu, bestimmte Lehrys, Schülys und Räume zu bestimmten Zeiten als nicht verfügbar zu markieren. Im Gegensatz zu externen Prüfungen werden vollständig geplante interne Prüfungen aber in der Bewertung berücksichtigt.

3.2. Einstellungen

Während ein Modell das Planungsproblem selbst beschreibt, enthalten die Einstellungen Parameter, wie die Planung ablaufen soll.

- Wie wichtig sind bestimmte Eigenschaften eines Plans? Ist es zum Beispiel wichtiger, dass ein Lehrly nur an wenigen Tagen für Prüfungen eingeteilt ist oder dass ein Lehrly an einem einzelnen Tag nicht sehr viele Prüfungen hintereinander hat?
- Wie viel Speicher und Rechenzeit soll für die Planung verwendet werden?
- ...

ACHTUNG: Bevor Sie Einstellungen, die die Bewertung von Plänen betreffen, ändern, sollten Sie genau verstanden haben, wie Planung, insbesondere Mutationen, sowie die Bewertung von Plänen funktioniert. Schlechte, insbesondere schlecht aufeinander abgestimmte Einstellungen, können dazu führen, dass keine guten Pläne gefunden werden.

3.3. Planung

Ein *Plan* ist eine Zuordnung eines Vorsitzendys, eines Protokollantys, eines Raums und einer Zeit zu jeder Prüfung. Prüfy und Schüly einer Prüfung sind unveränderlich und bereits im Modell festgelegt. Jeder Plan besitzt eine Bewertung, die angibt, wie gut oder schlecht ein Plan ist. Diese Bewertung besteht aus einem Zahlenwert, der aus dem Plan selbst, den Vorlieben des Modells sowie in den Einstellungen festgelegten Gewichten berechnet wird. Es ist schwer präzise festzulegen, was ein guter Plan ist und die Berechnung der Bewertung ist nicht einfach zu verstehen. Da diese jedoch zentral für die Planung ist und ein zumindest oberflächliches Verständnis dieser Bewertung wichtig für die effektive Nutzung von Prüfungsplaner ist, wird die Bewertung von Plänen unten in einem eigenen Abschnitt ausführlich erläutert.

3.3.1. Planpool

Eine Planung besteht hauptsächlich aus vielen erstellten Plänen, dem sogenannten *Planpool*. Ein Planpool besitzt eine Maximalgröße und ist in Gruppen, die ebenfalls eine Maximalgröße besitzen, unterteilt. Pläne einer Gruppe sind dabei aus dem gleichen Ursprungsplan entstanden und im Allgemeinen einander sehr ähnlich. Wird ein neuer Plan in den Planpool eingefügt, wird - sofern durch das Einfügen die Maximalgröße seiner Gruppe überschritten ist - zunächst der schlechteste Plan der Gruppe entfernt. Ist die Maximalgröße der Gruppe nicht überschritten, aber durch das Einfügen die Maximalgröße des gesamten Planpools, so wird der schlechteste Plan des Planpools entfernt. Dies kann bedeuten, dass der neue Plan auch selbst gelöscht wird, da er entweder zu schlecht für seine Gruppe oder den Planpool insgesamt ist.

3.3.2. manuelle Pläne

Zusätzlich zum Planpool gibt es Menge von *manuellen Plänen*. Diese sind von Nutzy explizit angelegte Pläne, die auch von Nutzy direkt bearbeitet werden können. Sie besitzen jeweils einen eindeutigen Namen. Im Gegensatz zu Plänen im Planpool werden manuelle Pläne nicht automatisch entfernt, wenn bessere Pläne gefunden wurden.

3.4. Überblick unterstützte Dateiformate

Intern werden von Prüfungsplaner sogenannte *Zustandsdateien* verwendet. Dies sind einzelne Binärdateien, die den gesamten Zustand, also Modell, Einstellungen und auch die Planung enthalten. Zustandsdateien können benutzt werden, um eine bereits gestartete Planung später fortzusetzen oder um ein Planungsproblem mit Kollegys zu teilen. Zustandsdateien sind jedoch nicht für die direkte Bearbeitung durch das Nutzy geeignet.

Für die direkte Bearbeitung durch das Nutzy werden zusätzliche *Modell-Verzeichnisse* als Import / Export Format angeboten. Dies sind Verzeichnisse, die das Modell und die Einstellungen als Textdateien enthalten. Modell-Verzeichnisse sind geeignet, um ein Problem initial zu beschreiben und langfristig zu speichern. Dateien können direkt vom Nutzy bearbeitet werden und es ist leicht möglich, Teile des Modells anderer Planungen zu übernehmen. Zudem sind die Dateien im Modell-Verzeichnis notfalls auch ohne Prüfungsplaner lesbar. Modell-Verzeichnisse enthalten keine Planung.

Fertige Pläne können schließlich als **CVS** oder **HTML** Datei exportiert werden. Die **CSV** Datei ist für maschinelle Weiterverarbeitung gedacht, während die **HTML** Datei für das Nutzy einfach lesbar ist. **CVS** Pläne können auch wieder von Prüfungsplaner importiert werden.

ACHTUNG: *Speichern* und *Öffnen* bezieht sich im Zusammenhang mit Prüfungsplaner immer auf Zustandsdateien. Modell-Verzeichnisse werden *Importiert* und *Exportiert*. Auch nach Änderungen in der GUI bspw. an Einstellungen wird ein Modell-Verzeichnis nicht automatisch aktualisiert. Hierfür ist ein expliziter Export nötig. Im Gegensatz dazu, gibt es eine Autospeichern Funktionalität, die in regelmäßigen Abständen eine Zustandsdatei speichert.

3.5. Modell-Verzeichnisse

Modell-Verzeichnisse sind Verzeichnisse, die einzelne Text-Dateien enthalten, die direkt vom Nutzy bearbeitet werden können. Die wichtigste dieser Dateien ist *pruefungen.csv*, die Informationen zu allen Prüfungen enthält. Alle anderen Dateien in einem Modell-Verzeichnis sind optional. Unterstützt werden folgende Dateien:

- **pruefungen.csv** (erforderlich) alle zu planenden Prüfungen, diese Datei kann aus LUST exportiert werden; eine Nachbearbeitung ist nur in Ausnahmefällen nötig
- **pruefungsextras.csv** Zusatzinformationen zu Prüfungen wie z.B. Einteilung in Prüfungsgruppen oder (Teil-)vorgaben für Planung
- **lehrys.yaml** Liste der Lehrys mit ihren Fächern
- **raeume.yaml** Liste der Räume mit Fächern, die darin geprüft werden können
- **zeiten.yaml** Zeiträume für Prüfungen
- **vorlieben.yaml** Vorlieben für die Planerstellung
- **settings.yaml** Einstellungen

Viele Teile eines Modells, wie z.B. die verwendeten Räume oder die Lehrys ändern sich kaum zwischen zwei Prüfungsjahren an einer Schule. Daher bietet es sich an, einzelne Dateien eines Modell-Verzeichnisses vom Vorjahr zu kopieren und über die grafische Oberfläche oder per Hand anzupassen.

Bei den **.yaml** Dateien handelt es sich um UTF-8 kodierte Yaml-Dateien. **YAML** ist eine einfache, relativ gut lesbare Auszeichnungssprache. **YAML**-Dateien können mit einem geeigneten Text-Editor direkt vom Nutzy bearbeitet werden. Ein einfacher Text-Editor wie das Windows Notepad genügt, komfortabler ist die Bearbeitung jedoch mit Editoren, wie Notepad++ oder VS Studio Code, die speziellen Support für **YAML** bereitstellen. Auch die grafische Oberfläche von Prüfungsplaner enthält einen einfachen Text-Editor mit grundlegender Unterstützung für **YAML**.

Die **.csv** Dateien sind UTF-8 kodierte **CSV**-Dateien. Für diese bietet sich zur Bearbeitung eine Tabellenkalkulation wie LibreOffice Calc oder Microsoft Excel an. **pruefungen.csv** kann aus LUST exportiert werden. Beachten Sie hierbei bitte, dass LUST **XLS** Dateien exportiert, die ins **CVS**-Format konvertiert

werden müssen. Diese ist leicht mit einer Tabellen-Kalkulation wie z.B. Libre-Office oder MS Excel möglich. Diese Programme können auch verwendet werden um `pruefungen.csv` manuell zu bearbeiten. Dies sollte jedoch nur in Ausnahmefällen nötig sein.

Während `pruefungen.csv` die Prüfungen selbst enthält, sollten alle evtl. gewünschten Zusatzinformationen in `pruefungenextras.csv` gespeichert werden. Bei diesen Zusatzinformationen handelt es sich um die Einteilung der Prüfungen in Gruppen, sowie um Vorgaben für die Planung, also bereits festgelegte Zeiten, Räume oder Lehrers für eine Prüfung. Es ist zwar möglich, `pruefungenextras.csv` händisch zu erstellen und zu bearbeiten. Eine Erstellung und Bearbeitung über die grafische Oberfläche - insbesondere mit Hinblick auf die Einteilung der Prüfungen in Gruppen - ist jedoch wesentlich komfortabler.

3.5.1. Demo-Version

In der Demo-Version von Prüfungsplaner werden Modelle direkt nach dem Import anonymisiert. Das bedeutet, dass die Namen von Schülern, Lehrern, Räumen und Klassen zufällig geändert werden. Zudem werden die Namen von Fächern zufällig miteinander vertauscht. Aus einer Prüfung von Max Mustermann bei Maria Musterfrau im Fach M kann so zu einer Prüfungen von Vorname 32 Nachname 33 bei LK23 im Fach BIO werden.

Die Anonymisierung belässt die Eingabe aus Planungssicht also unverändert. So kann Prüfungsplaner ausgiebig getestet werden. Findet Planungsplaner gute Pläne, so sollten auch für das nicht anonymisierte Modell solche Pläne gefunden werden, da ja nur die Namen geändert sind. Ein produktiver Einsatz der Demo-Version wird jedoch durch die Anonymisierung deutlich erschwert. Für den Produktiv-Einsatz erwerben Sie bitte eine Vollversion, die keine solche Anonymisierung durchführt.

Nicht von der Anonymisierung betroffen ist das Öffnen von Zustandsdateien. Es ist also möglich (und auch ein intendierter Anwendungsfall), dass ein Kollegium mit Vollversion ein Modell-Verzeichnis importiert, die Planung teilweise durchführt und dann seine Zustandsdatei mit Kollegiums teilt, die nur über die Demo-Version verfügen. Diese können dann weiter an der Planung arbeiten (z.B. erzeugte Pläne prüfen, weitere Vorlieben eintragen oder manuelle Änderungen an Plänen vornehmen).

3.6. Zustandsdatei

Zustandsdateien sind einzelne Binärdateien, die Modell, Einstellungen sowie den aktuellen Stand der Planung enthalten. Sie sind nicht für die direkte Bearbeitung durch das Nutzy gedacht. Statt dessen dienen sie dazu, den aktuellen Stand der Planung zu speichern und diese so einfach später oder auf einem anderen Rechner fortsetzen zu können. Die grafische Oberfläche erlaubt es, aus Zustandsdateien ein Modell-Verzeichnis sowie Pläne zu extrahieren. Speichern (und Autospeichern) aktualisiert immer nur die Zustandsdatei, nicht ein Modell-Verzeichnis, das evtl. importiert wurde.

Zustandsdateien sind nicht für langfristige Datensicherung geeignet, da sich das Binärformat zwischen Versionen von Prüfungsplaner ändern kann. Hier bieten sich statt dessen Modell-Verzeichnisse sowie der Export von Plänen an. Sie sind jedoch gut für die Zusammenarbeit mit Kollegiums geeignet.

Es wird garantiert, dass Versionen von Prüfungsplaner, die sich nur in der 4. Stelle der Versionsnummer unterscheiden, die gleichen Zustandsdateien verwenden können. So ist zum Beispiel Version 0.9.9.2 mit 0.9.9.4 garantiert kompatibel. Unterscheidet sich die Versionsnummer an anderen Stellen ist eine Kompatibilität in den allermeisten Fällen nicht gegeben. Im Gegensatz dazu können Modell-Verzeichnisse alter Versionen sowie als `CSV` Datei exportierte Pläne meist auch von neueren Versionen importiert werden.

3.7. Exportierte Pläne

Erstellte Pläne können als `CSV` sowie `HTML` exportiert werden. Die `CSV` Datei enthält alle Informationen und ist für den Import in LUST sowie sonstige automatisierte Weiterverarbeitung geeignet, inklusive dem Import der in Prüfungsplaner ein. Die `HTML` Dateien sind für die Begutachtung der Pläne durch das Nutzy gedacht. Sie enthalten Übersichten über wann welches Lehry eingeplant ist, die Raumbelegung, wie die Bewertung sich zusammensetzt, etc.

3.8. Intendierte Nutzung

Die angedachte Arbeitsweise mit den verschiedenen Ein- und Ausgabedateien ist wie folgt: Für jede durchzuführende Prüfungsplanung werden die Prüfungen als `XSL` Datei aus LUST exportiert und mittels

z.B. Microsoft Excel in `pruefungen.csv` umgewandelt. Andere Dateien des Modell-Verzeichnisses werden entweder aus vorherigen Planungen übernommen und angepasst oder neu erstellt. Die Erstellung bzw. Anpassung kann dabei über die grafische Oberfläche oder durch direkte Bearbeitung der `yaml` Dateien erfolgen.

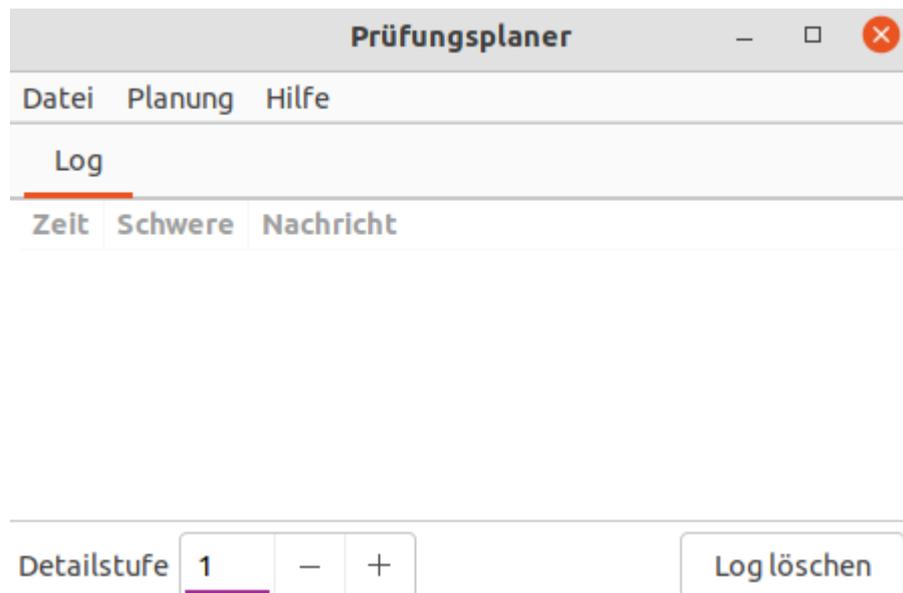
Während der Planung werden Zustandsdateien genutzt, um Zwischenstände zu speichern. Es wird davon ausgegangen, dass Zustandsdateien sich oft ändern. So wird die aktuelle Zustandsdatei durch die Autospeichern-Funktion bspw. automatisch überschrieben. Bei größeren Änderungen am Modell, z.B. der Eingabe von vielen Vorlieben, bietet es sich zur Datensicherung an, die neuen Informationen als Modell-Verzeichnis zu exportieren. Zusätzlich kann eine Sicherheitskopie der aktuellen Zustandsdatei erstellt werden. Zustandsdateien bieten sich auch an, um Planungen auf einem Rechner vorzubereiten und dann auf einem anderen Rechner auszuführen oder um Planungen mit Kollegys zu teilen.

Nach erfolgreicher Planung können erzeugte Pläne als `CVS` Datei exportiert und in `LUST` importiert werden. Da die `CVS` Dateien jedoch nur schlecht direkt vom Menschen lesbar sind, wird zusätzlich zu der exportierten `CVS` Datei auch eine für das Nutzy gedachte `HTML` Datei exportiert. Dieser `HTML` Export wird auch während der Planung benutzt, um Pläne zu begutachten.

4. Nutzung der grafischen Oberfläche

4.1. Laden eines Modells.

Nach dem Start der grafischen Oberfläche wird zunächst nur ein einzelner Reiter `Log` angezeigt.



Vor dem Beginn der eigentlichen Arbeit, muss zunächst ein `Modell` geöffnet werden. Hierzu kann über das Menü `Datei` entweder ein Modell-Verzeichnis importiert oder eine Zustands-Datei geöffnet werden. Nach dem Laden eines Modells stehen dann alle Reiter zur Verfügung.

Prüfungsplaner													
Datei Planung Hilfe													
Planung manuelle Planung Vorlieben Prüfungen Lehrlys Räume Zeiten Fächer Einstellungen Modell-Probleme Log													
Filter	Prüfy	Fach	Klasse	Art	Status	Auswahl		konfliktfrei					
<input checked="" type="checkbox"/> Gruppenansicht	Gruppen zuweisen			Gruppen klassenübergreifend zuweisen				Gruppen zurücksetzen					
Prüfy	Fach	Gruppe	Nachname	Vorname	Klasse	Stufe	Art	Protokollant	Vorsitzendy	Tag	Zeit	Raum	Planungsstatus
▼ LK1	REWE	- (1)											noch nicht geplant
└ LK1	REWE	-	Nachname 47	Vorname 47	KL1	GK	MP	-	-	-	-	-	noch nicht geplant
▼ LK1	REWE	Gruppe 1 (4)											noch nicht geplant
└ LK1	REWE	Gruppe 1	Nachname 1	Vorname 1	KL8	GK	MP	-	-	-	-	-	noch nicht geplant
└ LK1	REWE	Gruppe 1	Nachname 135	Vorname 135	KL9	GK	MP	-	-	-	-	-	noch nicht geplant
└ LK1	REWE	Gruppe 1	Nachname 82	Vorname 82	KL2	GK	MP	-	-	-	-	-	noch nicht geplant
└ LK1	REWE	Gruppe 1	Nachname 98	Vorname 98	KL7	GK	MP	-	-	-	-	-	noch nicht geplant
▼ LK1	REWE	Gruppe 2 (4)											noch nicht geplant
└ LK1	REWE	Gruppe 2	Nachname 109	Vorname 109	KL9	GK	MP	-	-	-	-	-	noch nicht geplant
└ LK1	REWE	Gruppe 2	Nachname 16	Vorname 16	KL3	GK	MP	-	-	-	-	-	noch nicht geplant
└ LK1	REWE	Gruppe 2	Nachname 3	Vorname 3	KL3	GK	MP	-	-	-	-	-	noch nicht geplant
└ LK1	REWE	Gruppe 2	Nachname 77	Vorname 77	KL10	GK	MP	-	-	-	-	-	noch nicht geplant
▼ LK10	E1	- (3)											noch nicht geplant
└ LK10	E1	-	Nachname 19	Vorname 19	KL10	GK	MP	-	-	-	-	-	noch nicht geplant
└ LK10	E1	-	Nachname 61	Vorname 61	KL2	GK	PR	LK22	LK28	23.06.2022	11:00	R7	extern geplant
└ LK10	E1	-	Nachname 90	Vorname 90	KL2	GK	PR	LK28	LK22	23.06.2022	12:00	R7	extern geplant
Änderungen verwerfen							Änderungen übernehmen						

4.2. Übersicht Reiter

Der Reiter *Planung* erlaubt das Starten und Stoppen der Planung sowie die Inspektion des aktuellen Planpools.

Der Reiter *manuelle Planung* erlaubt es Pläne genauer zu Betrachten, manuell zu Bearbeiten, zu Importierung und Exportierungen sowie einzelne Pläne automatisch zu Optimieren.

Unter *Prüfungen* können die zu planenden Prüfungen betrachtet, in Prüfungsgruppen eingeteilt und Planungen teilweise vorgegeben werden. Hier können die Teile des Modells, die in einem Modell-Verzeichnis in `pruefungen.csv` und `pruefungsextras.csv` gespeichert sind, bearbeitet werden.

Die Reiter *Vorlieben*, *Lehrlys*, *Räume*, *Zeiten* und *Einstellungen* erlauben das Betrachten und Bearbeiten der entsprechenden Teile des Modells sowie der Einstellungen. Die dargestellten Informationen entsprechen den Dateien `vorlieben.yaml`, `lehrlys.yaml`, `raeume.yaml`, `zeiten.yaml` sowie `einstellungen.yaml`. Diese Reiter verfügen jeweils über 2 Ansichten: eine grafische Ansicht und eine Text-Ansicht. Meist bietet sich wohl die Nutzung der grafischen Ansicht an, da diese intuitiver zu nutzen ist. Die Nutzung der Textansicht kann jedoch effizienter sein. Man kann zum Beispiel die Zeitvorlieben eines Raums in der grafischen Ansicht eintragen, dann in die Textansicht wechseln und diese Vorlieben schnell für mehrere andere Räume kopieren. Es ist in der Text-Ansicht auch einfach, verschiedene Einträge zu kombinieren. So könnte zum Beispiel ein Nutzer die Vorlieben aller Kollegys im Fachbereich A auf seinem Rechner eintragen, während ein anderer Nutzer die Vorlieben der Kollegys im Fachbereich B auf einem andere Rechner erfasst. Anschließend kann man beide Vorlieben in der Text-Ansicht einfach zusammenfügen.

Der Reiter *Fächer* zeigt an, welche Prüfungen, Lehrlys und Räume es für die einzelnen Fächer gibt. Dieser Reiter erlaubt kein Bearbeiten der Fächer, da diese implizit über Prüfungen, Lehrlys und Räume verwaltet werden. Die Fächer von Prüfungen können nicht geändert werden, die Fächer von Lehrlys und Räumen werden in den entsprechenden Tabs angepasst.

Wenn Probleme mit dem Modell bestehen, werden diese im Reiter *Modell-Probleme* angezeigt. Es wird zum Beispiel geprüft, ob für jede Prüfung mindestens ein Protokollant, Vorsitzendy, Raum und Zeitslot geeignet ist. Sind Modell-Probleme vorhanden, so ist keine automatische Planung möglich. Es werden viele, jedoch nicht alle Probleme automatisch gefunden. Selbst wenn keine Modell-Probleme gefunden werden, bedeutet dies nicht, dass ein konfliktfreier Plan existiert. Selbst wenn ein solcher existiert wird auch nicht garantiert, dass ein solcher Plan gefunden wird. Besonders Probleme, die sich aus der Kombination verschiedener Einschränkungen ergeben, werden nur schwer gefunden.

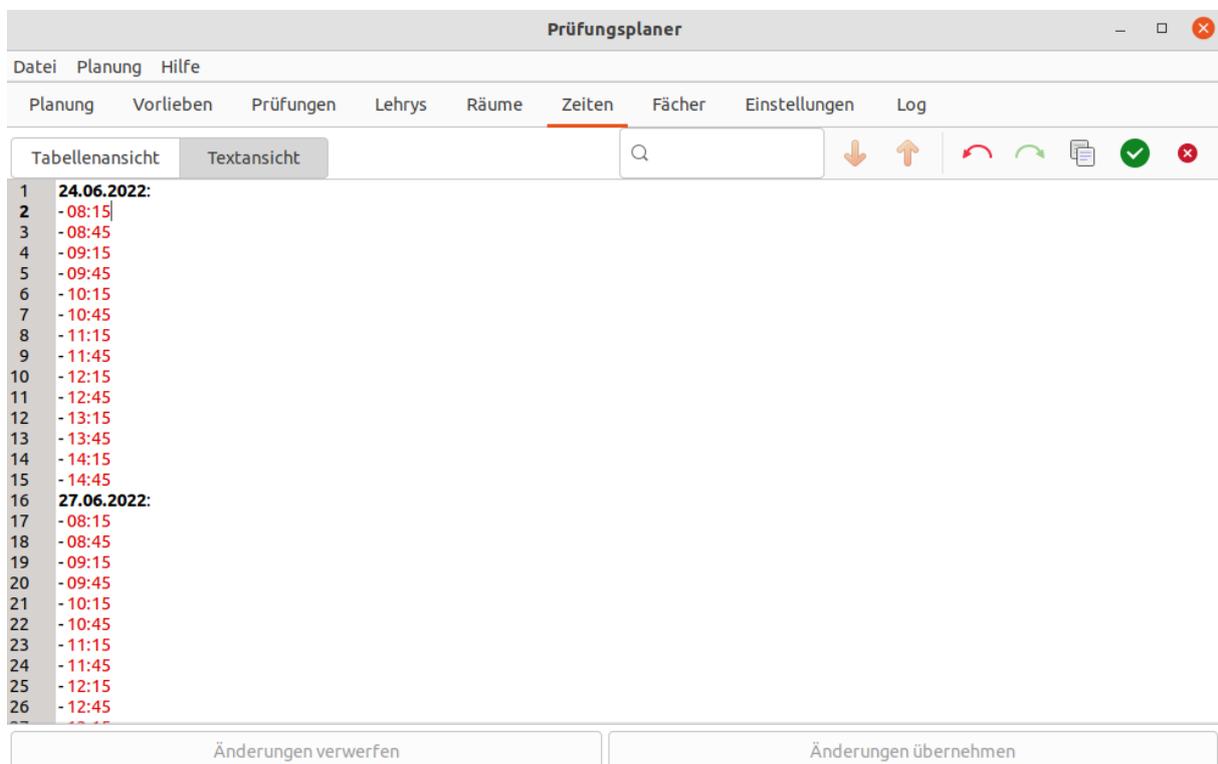
Der Reiter *Log* enthält ein Protokoll von wichtigen Ereignissen. Dieses Protokoll dient hauptsächlich der genauen Analyse des Verhaltens von Prüfungsplaner und insbesondere der Ausgabe von hoffentlich hilfreichen Zusatzinformationen bei Fehlern. Es kann eingestellt werden, wie detailliert das Protokoll sein soll. Bei Stufe 1 werden nur schwere Fehler angezeigt (die hoffentlich nie auftreten). Detailliertere Stufen erlauben es zum Beispiel zu protokollieren, wann welche Dateien gelesen oder geschrieben wurden und wie die Planung voranschreitet. Sehr hohe Detailstufen führen zusätzliche Tests aus, um eventuelle

Fehler in der Implementierung von Prüfungsplaner zu entdecken und führen daher zu einer deutlichen Verlangsamung der Planung.

4.3. Modell / Einstellungen bearbeiten

Die Bearbeitung des Modells sowie der Einstellungen erfolgt in jeweils 2 Stufen. Wird in einem der Reiter eine Änderung durchgeführt, so muss diese Änderung übernommen oder verworfen werden, bevor zu einem anderen Reiter gewechselt wird. Beim Übernehmen werden alle nötigen Änderungen am Modell, den Einstellungen und der Planung vorgenommen. Einige Änderungen, z.B. Änderungen an den Prüfungen, erfordern, dass eine evtl. bereits bestehende Planung verworfen (d.h. gelöscht) wird. Andere Änderungen, insbesondere an den Vorlieben oder Einstellungen erfordern kein Verwerfen einer bestehenden Planung, machen aber eine zeitaufwändige Neubewertung aller Pläne nötig.

Werden Änderungen in der Text-Ansicht vorgenommen, muss der Text neu eingelesen werden, um evtl. Änderungen zu erkennen. Dies geschieht automatisch beim Wechsel zwischen Text-Ansicht und grafischer Ansicht. Es kann explizit über einen Check-Button in der Toolbar ausgeführt werden. Es empfiehlt sich, die Texteingabe regelmäßig zu prüfen, um Probleme frühzeitig zu finden und damit einfacher beheben zu können. Über die Toolbar ist es auch möglich, Änderungen rückgängig zu machen, im Text zu suchen und den Text in die Zwischenablage zu kopieren.



4.4. Planung durchführen

Nachdem das Modell und die Einstellungen zur Zufriedenheit angepasst wurden, kann die Planung im Reiter *Planung* gestartet werden. Die Planung erfolgt in 2 Phasen: der Erzeugung zufälliger Pläne und der schrittweisen Verbesserung dieser zufällig erzeugten Pläne durch sogenannte *Mutationen*.

4.4.1. Erste Planungsphase: zufällige Generierung von Plänen

In der ersten Phase der Planung werden Plänen zufällig erzeugt. Das bedeutet, dass jeder Prüfung zufällig ein Vorsitzendy, ein Protokollant, ein Raum und eine Zeit zugewiesen wird. Eine naive Umsetzung würde zu sehr, sehr vielen extrem schlechten Plänen führen, in denen z.B. Lehrys oft gleichzeitig für mehrere Prüfungen eingeteilt sind. Daher werden die Wahrscheinlichkeiten während der zufälligen Generierung so gewählt, dass halbwegs gute Pläne wahrscheinlicher sind. So wird ein Raum zum Beispiel viel eher für eine Prüfung gewählt, wenn er zum entsprechenden Zeitpunkt noch frei ist. Dennoch sind die allermeisten generierten Pläne recht schlecht.

Nur die besten generierten Pläne werden für die Mutations-Phase benutzt. Um gute Gesamtergebnisse zu erzielen, ist es sinnvoll, sehr viel mehr Pläne zu erzeugen als in der nächsten Phase benutzt werden. Wie viele Pläne erzeugt und für die nächste Phase benutzt werden ist dabei konfigurierbar. Diese Konfiguration wird im Abschnitt zu Planungseinstellungen genauer erläutert.

4.4.2. Zweite Planungsphase: Mutationen

In der Mutations-Phase werden Pläne schrittweise durch Mutationen verbessert. Eine Mutation besteht aus einer Aneinanderreihung einer oder mehrerer Teilmutationen. Eine Teilmutation ist dabei eine einzelne, kleine Änderung an einem Plan. Beispiele für Teilmutationen sind, einen anderen Raum für eine Prüfung zu wählen oder eine Prüfung auf eine andere Zeit zu verschieben. Einige Teilmutationen betreffen auch ganze Prüfungsgruppen. So ist es zum Beispiel auch eine einzelne Teilmutation, das Vorsitzendy einer ganzen Prüfungsgruppe zu ändern.

Für schon recht gute Pläne führt eine erste Teilmutation meist zu einer Verschlechterung des Plans. Ändert eine Teilmutation beispielsweise den Raum einer Prüfung, so ist es recht wahrscheinlich, dass der neue Raum bereits belegt ist. Eine weitere Teilmutation könnte dann die in Konflikt stehende Prüfung in den gerade freigewordenen Raum legen. Für eine Mutation werden daher mehrere Teilmutationen hintereinander ausgeführt und der beste dabei entstandene Zwischenplan als Ergebnis der Mutation benutzt. Die maximale Anzahl der Teilmutationen, aus denen eine Mutation bestehen darf, ist konfigurierbar.

Die angewendeten Teilmutationen sind zufällig, aber nicht blind. Teilmutationen werden so gewählt, dass die Wahrscheinlichkeit den Plan zu verbessern recht hoch ist. Dafür wird die Bewertung des Plans betrachtet und solche Teilmutationen bevorzugt, die besonders schlecht bewertete Teile des Plans verändern. Wurde bereits erfolglos versucht, einen bestimmten Teil eines Planes zu bessern, wird die Wahrscheinlichkeit für Mutationen, die diesen Teil ändern, verringert.

Ein Problem ist, dass durch Mutationen sehr viele, sehr ähnliche Pläne erzeugt werden. Ohne geeignete Gegenmaßnahmen kommt es leicht dazu, dass der gesamte Planpool nur aus sehr ähnlichen Variationen eines einzigen Plans besteht. Dies würde dazu führen, einen Plan zu erzeugen, der zwar nicht einfach durch einzelne Mutationen verbessert werden kann, insgesamt gesehen aber dennoch relativ schlecht ist. Um dem entgegenzuwirken wird der Planpool in Gruppen von Plänen unterteilt. In einer Gruppe werden alle Pläne zusammengefaßt, die durch Mutationen aus dem gleichen Plan entstanden sind. Es ist konfigurierbar, wie viele Pläne eine Gruppe maximal enthalten darf, d.h. wie viele Varianten eines Planes maximal vorgehalten werden. Eine andere Gegenmaßnahme ist, immer mindestens eine gewisse Anzahl von Mutationen eines Planes zu erzeugen, bevor dieser Plan verworfen wird, weil nun bessere Pläne gefunden wurden.

4.4.3. Einstellungen für Planung

Eine wichtige Einstellung ist die Gesamtgröße des Planpools, d.h. die Anzahl der im Planpool vorgehaltenen und in jeder Mutations-Runde bearbeiteten Pläne. Große Planpools machen es wahrscheinlicher, auch global gute Pläne zu finden. Andererseits ist die für jede Mutations-Runde benötigte Zeit direkt proportional zur Planpoolgröße und der von Prüfungsplaner benötigte Arbeitsspeicher steigt bei großen Planpoolgrößen.

Eine andere wichtige, globale Einstellung ist die maximale Anzahl der Pläne pro Gruppe. Nur sehr wenige Pläne pro Gruppe führen dazu, dass eigentlich gute Pläne durch Mutationen leicht in Sackgassen geraten können. Viele Pläne pro Gruppe sorgen jedoch dafür, dass die Varianz innerhalb des Planpools relativ zu dessen Gesamtgröße recht gering ist und damit die Gesamtgröße des Pools recht groß gewählt werden muss.

Die letzte für die Planung wichtige, globale Einstellung ist die Anzahl der in der ersten Phase zufällig generierten Pläne. Diese Anzahl sollte viel höher, als die Planpoolgröße gewählt werden. So werden die meisten generierten Pläne direkt in der ersten Phase wieder verworfen und nur relativ gute Pläne schaffen es in die zweite Phase.

Prüfung außerhalb Gruppenzeit: Konflikt 1000 - + C

Planung

maximale Poolgröße:	150	-	+	C
maximale Gruppengröße:	5	-	+	C
Anzahl generierte Pläne:	2000	-	+	C
Anzahl Mutations-Runden:	0	-	+	C
Anzahl Mutationen pro Runde:	5	-	+	C
max. Anzahl Teilmutationen:	5	-	+	C
Autospeichern Intervall (s):	300	-	+	C
bearbeite nur beste Pläne:	0	-	+	C

Allgemeine Einstellungen

maximale Prüfungsgruppengröße:	4	-	+	C
--------------------------------	---	---	---	---

Änderungen verwerfen Änderungen übernehmen

Andere Einstellungen, die die Planung betreffen, können leicht während einer laufenden Planung geändert werden. Daher stehen diese auch direkt im Reiter *Planung* zur Verfügung. Dies sind vor allem die Einstellungen, aus wie vielen Teilmutationen eine Mutation maximal bestehen darf und wie viele Mutationen pro Plan pro Runde erstellt werden. Spät in der Planung, als wenn die Pläne bereits recht gut sind und nur noch schwer Verbesserungen gefunden werden können, kann es sinnvoll sein, die Anzahl der Teilmutation zu erhöhen. Eventfalls kann es dann sinnvoll sein, sich nur noch auf einige wenige beste Pläne (d.h. Plangruppen) zu konzentrieren.

4.4.4. Threads

Ein wichtiger Parameter, der aber nicht in Einstellungen gespeichert wird sondern nur im Reiter *Planung* zu finden ist, ist die Anzahl der zu benutzenden *Threads*. Threads sind gleichzeitig ausgeführte Teile des Programms. Viele Threads erhöhen zunächst die Gesamtverarbeitungszeit. Werden jedoch wesentlich mehr Threads verwendet als Prozessorkerne für deren Ausführung zur Verfügung stehen, sinkt die Gesamtgeschwindigkeit wegen erhöhtem Verarbeitungsaufwand wieder. Zudem kann es bei vielen Threads passieren, dass der Computer nur noch verzögert auf Eingaben reagiert. Es ist empfehlenswert, zwischen 2 und der Anzahl der zur Verfügung stehenden Prozessorkerne Threads zu verwenden. Als Standard-Wert wird die Anzahl der Prozessorkerne verwendet. Sollen gleichzeitig mit Prüfungsplaner viele oder rechenzeitintensive andere Anwendungen ausgeführt werden, empfiehlt es sich, eine niedrigere Threadanzahl zu verwenden.

Die Anzahl der zu verwendenden Threads ist für die Planung sowie die manuelle Optimierung von Plänen wichtig. Diese Einstellung betrifft jedoch die gesamte Ausführung von Prüfungsplaner. Es kann also passieren, dass auch die GUI wenig reaktiv wird, wenn eine nicht sinnvolle Anzahl von Thread konfiguriert wird. Es empfiehlt sich, nach Änderung der Threads eine evtl. laufende Planung zu stoppen und neu zu starten. So wird die Planung für die neue Anzahl von Threads angepasst.

4.4.5. Steuerung

In der unteren rechten Ecke, befindet sich ein Button, mit dem die Planung gestartet und gestoppt werden kann. Über diesem Button können einige Einstellungen geändert werden, die direkt während einer laufenden Planung Auswirkungen haben. Dort kann auch die Anzahl der zu verwendenden Threads geändert werden.

Prüfungsplaner (verändert)

Datei Planung Hilfe

Planung manuelle Planung Vorlieben Prüfungen Lehrlys Räume Zeiten Fächer Einstellungen Log

Plangruppe 63, KB: 10300, VB: -1005666, 11:07:25

Typ	Anzahl	Bewertung
> Mehrfache Raumbelegung	50	KB: 5000
> Lehrly-Raumwechsel mit Pause	43	KB: 4300
> Lehrly-Raumwechsel ohne Pause	2	KB: 1000
> 40 Prüfungen insgesamt	1	VB: 12000
> Pause der Länge 2	30	VB: 9000
> Pause der Länge 4	16	VB: 8800
> 36 Prüfungen insgesamt	1	VB: 8100
> 34 Prüfungen insgesamt	1	VB: 6450
> Pause der Länge 6	6	VB: 6000
> Pause der Länge 5	8	VB: 6000
> Pause der Länge 10	2	VB: 5000
> Pause der Länge 3	11	VB: 4400
> Pause der Länge 7	3	VB: 3900
> Vorliebe Raum / Zeit	15	VB: 3750
> Pause der Länge 11	1	VB: 3000
> 28 Prüfungen insgesamt	1	VB: 2700
> 22 Prüfungen insgesamt	3	VB: 2250
> Pause der Länge 9	1	VB: 2050
> 10 Prüfungen an einem Tag	4	VB: 2000

Details Aktualisieren Besten Plan anzeigen Bearbeiten

Starte Planung

Runde 58

Gruppe	Bester Plan	Schlechtester Plan	Größe
<input checked="" type="checkbox"/> 63	KB: 10300 VB: -1005666	KB: 10700 VB: -1009472	5
<input checked="" type="checkbox"/> 193	KB: 10600 VB: -1136843	KB: 10800 VB: -1135129	5
<input type="checkbox"/> 209	KB: 47600 VB: -1079027	KB: 55100 VB: -1077592	5
<input type="checkbox"/> 251	KB: 49300 VB: -1020015	KB: 55000 VB: -1004230	5
<input type="checkbox"/> 272	KB: 50000 VB: -878285	KB: 55600 VB: -889838	5
<input type="checkbox"/> 219	KB: 50100 VB: -1103035	KB: 51700 VB: -1078829	5
<input type="checkbox"/> 74	KB: 51700 VB: -1003504	KB: 57400 VB: -1010854	5
<input type="checkbox"/> 305	KB: 52100 VB: -1055622	KB: 52900 VB: -1082455	5
<input type="checkbox"/> 26	KB: 52900 VB: -1100727	KB: 54400 VB: -1100188	5
<input type="checkbox"/> 262	KB: 53600 VB: -1051753	KB: 58100 VB: -1046058	5

Threads: 7 - + C

max. Anzahl Teilmutationen: 5 - + C

Anzahl Mutationen pro Runde: 5 - + C

Anzahl Mutations-Runden: 0 - + C

Autospeichern Intervall (s): 300 - + C

Über diesen Einstellungen finden Sie einen Überblick über den aktuellen Stand der Planung. Sie sehen, welche Plangruppen es im Planpool gibt, wie der beste und schlechteste Plan jeder Gruppe bewertet ist und wie viele Pläne die Gruppen enthalten.

Durch Auswählen einer Gruppe wird der beste Plan dieser Gruppe links angezeigt. Es wird angezeigt, wie sich die Bewertung des Plans zusammensetzt. Diese Bewertung zu verstehen ist kompliziert. Bewertung werden detailliert in einem eigenen Abschnitt beschrieben. Buttons unter der Bewertung erlauben es, Details zum aktuellen Plan als HTML anzuzeigen, die Anzeige zu aktualisieren, d.h. den aktuell besten Plan der gleichen Plangruppe anzuzeigen sowie den aktuell besten Plan anzuzeigen. Zudem wird dort die Bearbeitung des aktuell angezeigten Planes ermöglicht. Diese Aktion speichert den aktuellen Plan als manuellen Plan, so dass er in Reiter *manuelle Planung* bearbeitet werden kann.

4.4.6. Export von Plänen

Mittels des Button *Details* kann der aktuell angezeigte Plan als HTML angezeigt werden. Dieses Format ist für die Betrachtung durch das Nutzy geeignet, jedoch nicht für einen Export, da es nur schlecht automatisch verarbeitet werden kann. Im Menü *Planung* ist es möglich einen oder mehrere Pläne als CSV Datei zu exportieren. Diese ist für den Import in LUST geeignet. Leider sind diese CSV Dateien schlecht geeignet, um einem Nutzy einen Überblick über den Plan zu verschaffen. Daher wird automatisch zu jeder exportierten CVS Datei auch eine zugehörige HTML Datei exportiert.

4.5. Manuelle Planung

Über den Reiter *manuelle Planung* ist es möglich, manuell Änderungen an Plänen vorzunehmen, interessante Pläne zu speichern, Pläne zu importieren und exportieren sowie einzelne Pläne automatisch zu optimieren.

4.5.1. Liste der manuellen Pläne

In der linken oberen Ecke findet sich eine Liste aller bekannten manuellen Pläne. Der üblichste Weg, einen Plan zu dieser Liste hinzuzufügen, ist die Bearbeitung eines während der Planung automatisch erzeugten Planes in Reiter *Planung*. Es ist jedoch auch möglich, einen Plan zu importieren, einen neuen Plan zu erstellen oder einen existierenden manuellen Plan zu duplizieren. Möchte man größere Änderungen an einem Plan vornehmen, kann es sinnvoll sein, zunächst eine Sicherungskopie des Originalplans mittels *Duplizieren* anzulegen.

4.5.2. Manuelle Bearbeitung

Im Hauptfenster des Reiters *manuelle Planung* zeigt den in der Liste der manuellen Pläne ausgewählten Plan an. Dort kann auch der Plan bearbeitet werden. Es können Protokollanty, Vorsitzendy, Tag, Zeit und Raum jeder Prüfung geändert werden. Prüfungen, die einer Gruppe angehören, werden unterhalb dieser Prüfungsgruppe angezeigt. Die Werte einer Gruppe bestehen für Lehrlys, Räume und Tage aus dem Wert der Prüfungen, wenn dieser für alle Prüfungen der Gruppe gleich ist, sowie der frühesten Zeit, falls alle Prüfungen am gleichen Tag geplant sind. Das Ändern des Wertes einer Gruppe ändert alle Prüfungen der Gruppe. Wird ein Wert geändert, wird der vorherige Wert in Klammern rechts neben dem neuen Wert angezeigt. Oberhalb der Liste der Prüfungen finden Sie die Bewertung des gesamten Plans, sowie die Änderung der Bewertung durch die letzte Änderung. Dort ist es auch möglich, die letzte Änderung rückgängig zu machen und wiederherzustellen. Ebenso kann die Liste der Prüfungen dort gefiltert werden, um einfacher einzelne Prüfungen zu finden.

4.5.3. Bewertungen

Links neben dem Plan wird unter der Liste der manuellen Pläne die detaillierte Gesamtbewertung des Plans ausgewählt. Ist eine Zeile im Plan ausgewählt, werden alle Bewertungen die die ausgewählte Prüfung bzw. Prüfungsgruppe betreffen ganz unten links angezeigt. Ist eine Prüfung bzw. Prüfungsgruppe an einem Konflikt beteiligt, wird die zugehörige Zeile im Plan rot markiert.

4.5.4. Automatische Optimieren / Einplanen

Ein typisches Szenario für den Einsatz der manuellen Planung ist, zunächst über die Planung einen guten Plan automatisch erzeugen zu lassen. Dieser wird dann manuell nachbearbeitet. Beispielsweise könnte das Nutzy bei der genaueren Prüfung des Plans feststellen, dass ein Lehry übermäßig oft eingeplant ist. Die Nachbearbeitung könnte dann darin bestehen, für einige Prüfungen dieses Lehrlys andere Lehrlys einzuplanen. Nachbearbeitungen führen oft zunächst zu einer Verschlechterung des Plans, ausserdem eröffnen Nachbearbeitungen oft Möglichkeiten für weitere Verbesserungen. Um einen manuellen zu verbessern, stehen die Funktionen *Optimieren* und *Einplanen* zur Verfügung.

Die Funktion *Optimieren* erlaubt es, den aktuellen manuellen mit den gleichen Methoden zu verbessern, die während der Mutationsphase der Planung verwendet werden. So können nach Änderungen Pläne schnell automatisiert optimiert werden. Im Gegensatz zur Mutationsphase wird nur ein einzelner Plan

optimiert. Es gibt keine Gruppe von ähnlichen Plänen und gefundene Verbesserungen werden sofort übernommen anstatt mehrere Mutationen durchzuführen und nur die beste Mutation zu übernehmen. Dies führt dazu, dass das automatische Optimieren Pläne zwar schneller als die Mutationsphase verbessert, dafür aber auch eher in lokalen Optima feststeckt. Nach Abschluss der Optimierung kann das Nutzy die vorgenommenen Änderungen begutachten und gegebenenfalls rückgängig machen.

Die Optimierung ist dafür geacht, um bereits gute Pläne schnell zu verbessern. Das Nutzy sollte die Ergebnisse der Optimierung prüfen. Sollen mehrere Pläne gleichzeitig mit relativ viel Rechenzeit und unter Vermeidung lokaler Optima optimiert werden, bietet es sich an, einen manuellen Plan *einzuplanen*, d.h. zum Planpool hinzuzufügen. So kann der Plan im Rahmen der normalen Planung verbessert werden.

5. Nutzung Kommandozeilen-Tool

Um das Kommandozeilentool einfach nutzen zu können, sollte es über die PATH-Variable zu finden sein. Anschließend started `pplan-cmd eingabe-verzeichnis` die Verarbeitung des Planungsproblems, das im Verzeichnis `eingabe-verzeichnis` gespeichert ist. Es wird erwartet, dass dieses Eingabe-Verzeichnis ein Modell-Verzeichnis ist. Ist dort zusätzlich eine Zustandsdatei `state.ppl` zu finden, so wird diese statt des Modell-Verzeichnisses geladen. So ist es möglich, bereits gestartete Bearbeitungen fortzusetzen. Nach der in `settings.yaml` festgelegten Anzahl von Mutationsrunden beendet sich das Tool automatisch und exportiert die besten Pläne. Während der Ausführung können über Tastenkürzel Aktionen ausgeführt werden:

- `q` Programm beenden
- `x` Programm beenden ohne Speichern
- `s` Sichere aktuellen Zustand in `state.ppl`
- `e` Exportiere Zwischenstand (5 beste Pläne)
- `E` Exportiere Zwischenstand (5 schlechteste Pläne)
- `h` Hilfe anzeigen

Weitere Kommandozeilenoptionen erlauben es unter anderem, eine bereits angefangene Bearbeitung neu zu starten, zusätzliche Mutationsrunden auszuführen, nur die besten Pläne zu bearbeiten etc. Die Liste aller Optionen erhalten Sie mittels `pplan-cmd --help`.

Diese Liste enthält jedoch nicht Parameter des Haskell-Runtime-Systems. Diese können nach dem Parameter `+RTS` angegeben werden. Die am meisten benutzte dieser Optionen ist vermutlich die Wahl der Anzahl der zu verwendenden Threads. Dies geschieht mittels `+RTS -N n` wobei `n` die Anzahl der Threads ist. Für das Parsen der Kommandozeilenparameter wird intern das Haskell-Paket `optparse-applicative` verwendet. Dieses bietet verschiedene Hilfestellungen für Autovervollständigung an. Das versteckte Argument `--bash-completion-script` erzeugt zum Beispiel ein Script, dass in der Bash-Shell für Autovervollständigung genutzt werden kann. Für eine Beschreibung dieser Feature siehe bitte die Dokumentation von `optparse-applicative`.

6. Bewertung von Plänen

Zentral für das Verständnis von *Prüfungsplaner* ist, wie Pläne bewertet werden. Grundsätzlich wird zwischen *Konflikten* und *Vorlieben* unterschieden. Ein Konflikt ist ein schweres Problem, das die Umsetzung eines Plans verhindert. Beispielsweise kann kein Lehry gleichzeitig an zwei Orten sein. Vorlieben drücken Wünsche aus. Auch wenn ein Plan eine schlechte Vorliebenbewertung hat, kann er umgesetzt werden. Ein Beispiel für eine Vorliebe ist, dass ein Lehry keine langen Pausen zwischen Prüfungen haben sollte.

Die Bewertung eines Planes besteht aus zwei Zahlen: einer Konfliktbewertung (KB) und einer Vorliebenbewertung (VB). Kleinere Werte sind hierbei besser. Der KB Wert ist immer größer oder gleich 0. Ein KB Wert von 0 bedeutet, dass kein Konflikt vorliegt. VB kann positiv oder negativ sein. Beim Vergleich von Bewertungen zweier Pläne werden zunächst die KB Werte verglichen. Nur wenn diese gleich sind, werden die VB Werte betrachtet.

Die Bewertung eines Planes setzt sich aus einer Summe von Einzelbewertungen zusammen. Einzelbewertungen sind immer reine Konflikt- oder reine Vorliebenbewertungen. Im Folgenden werden diese Einzelbewertungen beschrieben. Die Gewichte der Einzelbewertungen können in `settings.yaml` im Abschnitt

Bewertungen konfiguriert werden. Einige Einzelbewertungen sind dabei zwingend Konfliktbewertungen, während andere als Vorliebe oder Konflikt konfiguriert werden können.

6.1. Warnung vor unvorsichtigen Einstellungsänderungen

Wie Pläne bewertet werden, ist kompliziert und es erfordert Zeit und Aufwand ein Verständnis hierfür zu erwerben. Ein solches Verständnis ist aber essentiell, bevor Sie die Gewichte der Einzelbewertungen ändern. Schlecht aufeinander abgestimmte Einstellungen können dazu führen, dass keine guten Pläne erzeugt werden können. Weniger kritisch ist das Erstellen von Vorlieben. Dennoch sollten Sie auch hierfür ein Verständnis der Bewertung besitzen, um unliebsame Überraschungen zu vermeiden.

Eine Mutation besteht aus wenigen, direkt hintereinander ausgeführten Teilmutationen. Es gibt viele unterschiedliche Teilmutation, für die Wahl guter Einstellungen genügt es zu wissen, dass eine Teilmutation üblicherweise ein Planungsmerkmal einer Prüfung oder einer Prüfungsgruppe ändert. Teilmutation kann beispielsweise den Raum aller Prüfungen einer Prüfungsgruppe oder die Zeit einer einzelnen Prüfung ändern. Im Allgemeinen ändert eine Teilmutation nicht mehrere Merkmale oder viele Prüfungen. Es ist extrem wichtig, dass durch wenige Teilmutationen ein Plan besser werden kann. Große Verbesserungen müssen über Zwischenschritte, also mehrere kleine Verbesserungen erreichbar sein.

6.1.1. Beispiel

Pläne sollten nicht unnötig lange Pausen für Lehrers enthalten. Prüfungsplaner erlaubt es Pausen nach Ihrer Länge (in Zeitfenstern) zu bewerten. Eine naive (nicht wirklich schlechte, aber auch keine gute) Einstellung wäre z.B. $50 \times \text{Länge}$. Betrachten wir nun folgendes Beispiel: Es finden an einem Tag 3 Blöcke von Prüfungen statt, die ein Lehry involvieren: P1, P2, P3. Zwischen P1 und P2 liegt eine Pause der Länge L1 und zwischen P2 und P3 eine Pause der Länge L2. Die Bewertung hiervon wäre also $50 \times L1 + 50 \times L2$. Pausenzeiten sollen minimiert werden. Wird P3 um c Zeitfenster früher eingeplant, klappt auch alles, und wir erhalten ein Gewicht von $50 \times L1 + 50 \times (L2 - c)$. Bei der Pause zwischen P1 und P2 ist es jedoch schwieriger. Angenommen wir verschieben P2 um c Zeitfenster. Dann wird die Pause zwischen P2 und P1 zwar kleiner, die Pause zwischen P2 und P3 aber größer: $50 \times (L1 - c) + 50 \times (L2 + c)$. Die Gesamtbewertung bleibt also gleich. Für eine Verbesserung sind 2 Teilschritte nötig: erst P2 verschieben und danach P3. Die implementierten Mutationen führen diese 2 Schritte aus; die Wahrscheinlichkeit hierfür ist aber wesentlich geringer, als das Verschieben von P2 und P3 einzeln zu finden. In der Praxis bedeutet dies, dass die Pläne langsamer optimiert werden, als wenn die Zwischenschritte bereits Verbesserung bedeuten.

Wir hätten also gerne Pausenbewertungen, die idealerweise folgende Kriterien erfüllen:

- kurze Pausen sind besser als lange
- eine lange Pause ist besser als mehrere kleine Pausen gleicher Gesamtlänge
- für Pausenlängen $L1$, $L2$, Verschiebung c , und Pausenbewertung B sollte idealerweise gelten: $B(L1) + B(L2) > B(L1-c) + B(L2+c)$. (Dies ist nicht immer möglich, z.B. für $L1 := 2$, $L2 := 1$, $c := 1$).

Dies wird zum Beispiel durch folgende Einstellungen halbwegs erreicht, die mit einem Basiswert von 150 beginnt, diesen dann initial um 100 erhöht, wobei die Schrittweite aber immer um 5 verringert wird, also $0 : 0$, $1 : 150$, $2 : 250$, $3 : 345$, $4 : 435$, ...

Für $L1 := 3$, $L2 := 2$ und $c = 2$ ergibt sich zum Beispiel:

$$B(L1) + B(L2) = B(3) + B(2) = 345 + 250 = 595 >$$

$$B(L1-c) + B(L2+c) = B(1) + B(4) = 150 + 435 = 585$$

6.1.2. Negativbeispiel

Ein Nutzy möchte ungedingt Pläne erhalten, in denen ein Lehry alle Prüfungen eines Tages in einem Raum hat. Ein Raumwechsel ist also selbst mit Pausen nicht nötig. Das Nutzy ändert daher die Einstellung

Lehry Raumwechsel mit Pause von einer Vorliebe auf einen Konflikt und ändert den Wert auf KB 500. Gleichzeitig ist ein Raum-Konflikt, also eine Doppelbelegung eines Raumes, mit KB 100 bewertet. Diese Einstellungen führen dazu, dass Pläne erzeugt werden, die wenig Raumwechsel, dafür aber viele Raum-Konflikte enthalten. Dies ist selbst dann der Fall, wenn Pläne ohne Raumwechsel und Raum-Konflikte existieren. Das Problem ist, dass Zwischenschritte zu deutlichen Verschlechterungen führen und viele geeignete Teilmutationen direkt hintereinander für eine Verbesserung nötig ist.

Angenommen, ein Lehry hat einem Tag zunächst Prüfungen in Raum R1 und dann (mit Pause) in Raum R2. Diese Situation führt zu einer Bewertung von KB 500 für den Raumwechsel. Werden nun die Prüfungen in R2 in Raum R1 umgelegt führt dies wahrscheinlich zu einer Verbesserung, selbst wenn Raum R1 bereits belegt ist (andere Lehrys und andere Bewertungen mal ausgenommen), da wahrscheinlich weniger als 5 Raumkonflikte entstehen und ein Raumkonflikt mit KB 100 bewertet wird. Es ist also recht wahrscheinlich Raumkonflikte einzuführen.

Umgekehrt, angenommen alle Prüfungen eines Lehrys an einem Tag finden in Raum R1 statt. Es gibt aber einen Raum-Konflikt für einige der Prüfungen. Werden diese nun in Raum R2 umgelegt, so entstehen wahrscheinlich 2 Raumwechsel mit Pause: der Wechsel in R2 und zurück nach R1. Da vermutlich weniger als 10 Prüfungen in Konflikt standen, führt dies zu einer deutlichen Verschlechterung und die Raumkonflikte bleiben erhalten.

6.1.3. Generelle Empfehlungen

Einstellungen nach dem Prinzip *viel hilft viel* anzupassen, führt - wie obiges Negativbeispiel zeigt - oft zu Problemen. Dies gilt auch für Vorlieben. Ich empfehle mit den Standardeinstellungen zu starten, einige schnelle Probleplanungen durchzuführen und die resultierenden Pläne genau zu untersuchen. Je nach Ergebnis dieser Untersuchung können die Einstellungen **vorsichtig** angepasst werden. Evtl. helfen auch Experimente mit manueller Planung dabei, ein Verständnis für die Bewertung zu erhalten.

6.2. Implizite Konflikte

Für manche grundlegenden Konflikte gibt es keine Bewertung, da Prüfungsplaner diese **immer** bei der Planung vermeidet. Dabei handelt es sich um ganz einfache, grundlegende Bedingungen:

- Prüfungen werden nur in für das Fach geeigneten Räumen geplant.
- Als Vorsitzendy oder Protokollant werden nur Lehrys gewählt, die das Fach unterrichten.
- Prüfungen werden nur zu in den Eingaben genannten Zeiten geplant.
- Eine eventuell bereits vorhandene Planung von Prüfungen wird nicht geändert.

Auch über die manuelle Planung ist es nicht möglich, solche Konflikte einzuführen.

6.3. Ressourcenkonflikte

Ressourcenkonflikte sind Konflikte, die die mehrfache Einplanung einer *Resource* beschreiben, die nur einmal zu Verfügung steht. Ressourcen sind hierbei Räume, Lehrys und Schülys. Ressourcenkonflikte werden im Unterabschnitt **RessourcenBewertungen** konfiguriert.

6.3.1. Mehrfache Raumbelegung

Ein Raum kann nur für eine Prüfung gleichzeitig genutzt werden. Dies ist zwingend ein Konflikt. Die Schwere des Konflikts kann über die Einstellung **RaumDoppeltKonflikt** konfiguriert werden.

6.3.2. Lehry mehrfach eingeplant

Ein Lehry kann nur an einer Prüfung gleichzeitig teilnehmen. Es ist hierbei egal, ob ein Lehry als Prüfy, Vorsitzendy oder Protokollant an einer Prüfung teilnimmt. Verletzungen dieser Regel sind zwingend ein Konflikt. Die Schwere des Konflikts kann über die Einstellung **LehryDoppeltKonflikt** konfiguriert werden.

6.3.3. Prüfy mehrfach eingeplant

Es ist schwerer einen Konflikt von mehrfach eingeplanten Lehrys zu beseitigen, wenn es sich um das Prüfy handelt. Deswegen gibt es zusätzlich zu den *Lehry mehrfach eingeplant* Konflikten einen Konflikt speziell

für Prüfys. Die Schwere des Konflikts kann über die Einstellung `PruefyDoppeltKonflikt` konfiguriert werden.

6.3.4. Schüly hat mehrere Prüfungen an gleichem Tag

Ein Schüly kann maximal eine Prüfung pro Tag ablegen. Verletzungen dieser Regel sind zwingend ein Konflikt. Die Schwere des Konflikts kann über die Einstellung `SchuelyTagKonflikt` konfiguriert werden.

6.4. Lokale Einzelbewertungen

Lokale Einzelbewertungen sind Einzelbewertungen, die für die Planung einer einzelnen Prüfung ohne Kenntnis der Planung anderer Prüfungen erstellt werden können. Beispiele sind:

- Lehry kann an einem bestimmten Tag nur ungern an Prüfungen teilnehmen
- Kollegys arbeiten gern zusammen
- Raum steht an einem Tag nicht zur Verfügung

Lokale Einzelbewertungen können in der Eingabe-Datei `vorlieben.yaml` konfiguriert werden. Dort kann für Kombinationen einzelner Elemente des Plans einer Prüfung jeweils eine Einzelbewertung eingestellt werden. Es kann jeweils einer der Werte `...`, `---`, `--`, `-`, `o`, `+`, `++`, `+++`, `...` oder `x` eingestellt werden. `o` bedeutet, dass diese Kombination weder gut noch schlecht ist; dies ist der Vorgabewert, wenn nichts explizit eingetragen wird. `x` verbietet die Kombination; sie wird als Konflikt gewertet. Bei den anderen Werten, wird die Kombination als Vorliebe gewertet, `-` ist eine negative Vorliebe, `+` eine positive Vorliebe, je mehr Zeichen kombiniert werden, umso stärker.

Das Gewicht der Bewertung ergibt sich aus einem Basiswert, der für jede Art von Kombination in `settings.yaml` im Abschnitt `Bewertungen/LokaleBewertungen` konfiguriert werden kann. Für `x` wird der dort eingestellte Konfliktwert gewählt. Für Vorlieben wird die Anzahl der `-` oder `+` Zeichen gezählt und exponentiell mit Basis 5 bewertet. `--` zählt zum Beispiel als $(5^{(2-1)}) = 5$ multipliziert mit dem Basiswert. `++++` zählt als $-(5^{(4-1)}) = -125$ und `-` als $5^{(1-1)} = 1$.

Folgende Kombinationen von Elementen des Plans einer einzelnen Prüfung können bewertet werden:

Kombination	Einstellung	Beschreibung
Zeit	<code>Zeit</code>	Prüfungen können generell gern / ungern zu Zeit stattfinden; zum Beispiel vermeide Prüfungen spät am Nachmittag
Raum / Zeit	<code>RaumZeit</code>	Raum kann zu Zeitpunkt gern / ungern genutzt werden
Raum	<code>Raum</code>	Prüfungen können generell gern / ungern in Raum stattfinden
Lehry / Zeit	<code>LehryZeit</code> , <code>PruefyZeit</code>	Lehry kann zu bestimmten Zeiten gern / ungern an Prüfungen teilnehmen; Prüfys können schwerer umgeplant werden und haben daher eine eigene Gewichtung.
Lehry / Raum	<code>LehryRaum</code>	Lehry benutzt gern / ungern bestimmten Raum
Lehry / Schüly	<code>LehrySchuely</code>	Lehry prüft gern / ungern Schüly; es werden nur Protokollantys und Vorsitzendys berücksichtigt, da das Prüfy nicht geändert werden kann.
Lehry / Lehry	<code>LehryLehry</code>	Kollegys arbeiten gern / ungern zusammen. Es ist egal ob die Lehrys als Prüfy, Protokollanty oder Vorsitzendy an der Prüfung teilnehmen.
Lehry	<code>Lehry</code>	Lehry generell gern / ungern einplanen
Schüly / Raum	<code>SchuelyRaum</code>	Schüly wird gern / ungern in Raum geprüft
Schüly / Zeit	<code>SchuelyZeit</code>	Schüly hat Prüfungen gern / ungern zu bestimmter Zeit

6.4.1. Empfehlungen

Die Standard-Einstellungen sind auf die Vorlieben `-` und `+` ausgelegt. Stärker gewichtete Vorlieben wie `--`, `---`, `...` sollten nur mit Bedacht benutzt werden.

Die Bedeutung von Vorlieben kann schnell missverstanden werden. Angenommen ein Lehry hält gern Prüfungen in einem bestimmten Raum ab. Erstellt man eine `Lehry / Raum` Vorliebe `+` so könnte man leicht davon ausgehen, dass dies bedeutet: *Wenn das Lehry an der Prüfung beteiligt ist, versuche die Prüfung in*

den Raum zu legen. Tatsächlich deutet diese Vorliebe jedoch, dass die Kombination von Lehry und Raum vor allen anderen Lehry / Raum Kombinationen bevorzugt wird. Das schliesst auch Kombinationen mit anderen Lehrlys ein. Dies führt dazu, dass das Lehry vermutlich für mehr Prüfungen als ohne die Vorliebe eingeteilt wird.

Aus diesen Gründen kann es sinnvoll sein, ausschließlich negative Vorlieben zu nutzen und diese sparsam einzusetzen.

6.5. Lehry Einzelbewertungen

Lehry Einzelbewertungen sind Bewertungen, die sich aus dem Prüfungsplan für ein Lehry ergeben. Sie können in `settings.yaml` im Abschnitt `Bewertungen/LehryBewertungen` konfiguriert werden.

6.5.1. Anzahl Prüfungstage

Statt an vielen Tagen an wenigen Prüfungen teilzunehmen, bevorzugen es viele Lehrlys, die Prüfungen an wenigen Tagen zu konzentrieren. Für jede Anzahl von Prüfungstagen wird eine eigene Bewertung konfiguriert. Die Idee ist, dass es relativ egal ist, ob ein Lehry an 3 oder 4 Tagen Prüfungen hat. Es ist wichtiger, ob ein Lehry an keinem Tag Prüfungen hat (also nicht an den mündlichen Abiturprüfungen teilnimmt) oder an einem Tag.

Die Optimierung dieser Einzelbewertung kann dazu führen, dass für ein Lehry evtl. sehr viele Prüfungen an einem Tag geplant werden oder dass wenige Lehrlys insgesamt sehr viele Prüfungen haben, während Kollegys kaum eingeplant sind. Dieser Tendenz sollte über entsprechende Bewertungen der Anzahl von Prüfungen pro Tag und der Gesamtzahl der Prüfungen entgegengewirkt werden.

Es ist relativ schwer (d.h. unwahrscheinlich), die Anzahl der Prüfungstage eines Lehrlys in nur einer Mutation zu verringern. Es kann daher sinnvoll sein, bereits Zwischenschritte positiv zu werten. Hat ein Lehry zum Beispiel an einem Tag 4 Prüfungen und an einem anderen Tag 3 Prüfungen, so ist es relativ schwer, in einer einzigen Mutation 3 Prüfungen auf einen anderen Tag zu legen ohne den Plan anderweitig zu verschlechtern. Wird ein Plan, der an einem Tag 5 Prüfungen und an einem anderen nur 2 Prüfungen hat, bereits besser und ein Plan, der 6 / 1 Prüfungen besitzt, sogar noch besser bewertet, so sind Zwischenschritte möglich. Es werden dann eher Pläne mit wenigen Prüfungstagen erzeugt. Dies kann durch geschickte Wahl der Bewertungen der Anzahl der Prüfungen eines Lehrlys an einem Tag erreicht werden.

Andererseits ist es den meisten Lehrlys wahrscheinlich lieber 4 und 3 Prüfungen zu haben, anstatt 6 und nur eine Prüfung. Dies ist daher ein gutes Beispiel dafür, dass das Nutzy abwägen muss, welche Eigenschaften eines Planes dem Nutzy wie wichtig sind.

6.5.2. Viele Prüfungen pro Tag

Sehr viele Prüfungen an einem Tag sollten vermieden werden. Für jede Anzahl von Prüfungen pro Tag kann eine eigene Bewertung über `LehryVielePruefungenAnTag` hinterlegt werden.

6.5.3. Viele Prüfungen

Sehr viele Prüfungen insgesamt sollten vermieden werden. Für jede Anzahl von Prüfungen kann eine eigene Bewertung über `LehryVielePruefungen` hinterlegt werden. Dies kann benutzt werden, um die Prüfungen halbwegs gleichmäßig unter den Kollegys zu verteilen.

6.5.4. Lange Pausen

Lange Pausen zwischen Prüfungen sollten vermieden werden. Für jede Länge von Pausen kann eine eigene Bewertung in `settings.yaml` als `LehryPausen` hinterlegt werden. Es wird empfohlen, Pausen der Länge 1 nur leicht negativ zu werten, da solche kurzen Pausen für z.B. Raumwechsel und zur Pufferung von Prüfungen, die länger als erwartet dauern, wichtig sind. Dennoch ist es sinnvoll, auch Pausen der Länge 1 leicht negativ zu werten, da sonst leicht unnötige Pausen in Plänen erzeugt werden.

6.5.5. Viele Prüfungen ohne Pause

Viele Prüfungen eines Lehrlys am Stück sollten vermieden werden. Dies dient der Pufferung von unerwartet langen Prüfungen. Für jede Anzahl von Prüfungen kann eine eigene Bewertung in `settings.yaml` als `LehryPruefungenOhnePause` hinterlegt werden. Es wird empfohlen zumindest alle Prüfungen einer Gruppe ohne negative Bewertung zuzulassen.

Diese Einzelbewertung hat ähnliche Effekte wie Einzelbewertungen für *Lehry / Prüfy - Raumwechsel ohne Pause*. Diese Einzelbewertung greift jedoch auch, wenn die Prüfungen alle im gleichen Raum stattfinden.

6.5.6. Lehry / Prüfy - Raumwechsel ohne Pause

Vor dem Wechsel eines Lehrlys in einen Raum sollte das Lehry eine Pause haben. Dabei ist es wichtiger, dass das Prüfy eine Pause hat als dass das Protokollanty oder das Vorsitzendy eine Pause hat. In `settings.yaml` kann eine Bewertungen für Prüfys (`PruefyRaumwechselOhnePause`) und eine gemeinsame Bewertung für Protokollantys und Vorsitzendys (`LehryRaumwechselOhnePause`) hinterlegt werden. Bleibt ein Lehry im gleichen Raum, wechselt aber sein Rolle (ist also zum Beispiel für eine Prüfung Prüfy und die nächste Prüfung Protokollanty), wird dies nicht als Raumwechsel gewertet.

Die Pause erlaubt es, ohne Hast Räume zu wechseln und Vorbereitungen treffen zu können. Dafür muss auch der Raum, in den gewechselt wird, vorher frei sein. Dies wird über eine Raum-Einzelbewertung bewertet.

6.5.7. Lehry Raumwechsel mit Pause

Idealerweise sollte auch bei einer Pause ein Lehry den Raum nicht wechseln müssen und der Raum in der Pause des Lehrlys frei sein. So kann das Lehry sich länger im Raum aufhalten und Unterlagen im Raum gelagert lassen. Die Bewertung für Raumwechsel mit Pause kann in `settings.yaml` als `LehryRaumwechselMitPause` konfiguriert werden.

6.6. Prüfungsgruppe Einzelbewertungen

Für Prüfungen innerhalb einer Gruppe kann das gleiche Thema verwendet werden. Daher müssen die Prüfungen einer Gruppe innerhalb eines engen Zeitfensters geplant werden. So kann durch Beaufsichtigung vor und nach den Prüfungen sichergestellt werden, dass sich Schüllys nicht über gegenseitig helfen können.

6.6.1. Prüfung außerhalb Gruppenzeit

Eine außerhalb des Zeitfensters für ihre zugehörige Gruppe geplante Gruppe ist zwingend ein Konflikt. Die Schwere dieses Konflikts kann in `settings.yaml` als `PruefungAusserhalbGruppenZeitKonflikt` konfiguriert werden. Dies ist ein sehr seltener Konflikt, der nur von Mutationen vermieden wird. Der Konflikt kann aber bei manueller Planung auftreten oder wenn die Prüfungsgruppen geändert werden, während eine Planung bereits im Gange ist.

6.6.2. Mehrere Vorsitzendys / Protokollantys

Nach Möglichkeit sollte für alle Prüfungen einer Gruppe das selbe Vorsitzendy und das selbe Protokollanty eingeplant werden. Dies kann als Vorliebe oder Konflikt in `settings.yaml` als `PruefungsgruppeMehrereProtokollantys` bzw `PruefungsgruppeMehrereVorsitzendys` konfiguriert werden.

6.7. Raum Einzelbewertungen

Raum Einzelbewertungen sind Bewertungen, die sich aus dem Prüfungsplan für einen Raum ergeben. Sie können in `settings.yaml` im Abschnitt `Bewertungen/RaumBewertungen` konfiguriert werden.

6.7.1. Anzahl Prüfungstage

Wenn möglich sollte ein Raum an wenigen Tagen genutzt werden. So wird der normale Unterrichtsablauf weniger eingeschränkt. Für jede Anzahl von Tagen, an denen der Raum für Prüfungen genutzt wird, kann in `RaumAnzahlPruefungstage` eine eigene Bewertung konfiguriert werden.

6.7.2. Viele Prüfungen ohne Pause

Viele Prüfungen in einem Raum am Stück ohne Pause sollten vermieden werden. Dies dient der Pufferung von unerwartet langen Prüfungen. Für jede Anzahl von Prüfungen kann eine eigene Bewertung in `settings.yaml` als `RaumPruefungenOhnePause` hinterlegt werden. Es wird empfohlen, zumindest alle Prüfungen einer Gruppe ohne negative Bewertung zuzulassen.

6.7.3. Lehry- / Prüfy-Wechsel ohne Pause

Vor dem Wechsel eines Lehrys in einen Raum sollte der Raum frei sein. In `settings.yaml` können Bewertungen für Prüfys (`RaumPruefyWechselOhnePause`) sowie andere Rollen (`RaumLehryWechselOhnePause`) hinterlegt werden.