

39. Bundeswettbewerb Informatik, 1. Runde

Jan Ehehalt, Jonathan Hager

Inhaltsverzeichnis

1	Wörter aufräumen	3
1.1	Lösungsidee	3
1.2	Umsetzung	3
2	Dreieckspuzzle	3
3	Tobis Turnier	3
4	Streichholzrätsel	3
5	Wichteln	3
5.1	Lösungsidee	3
5.2	Umsetzung	4

1 Wörter aufräumen

1.1 Lösungsidee

Es werden zwei Listen angelegt, die jeweils alle Lücken und alle vollständigen Wörter getrennt speichern. Das Programm geht die Liste der Lücken durch und fügt für jede Lücke, der sich nur ein Wort eindeutig zuordnen lässt, das entsprechende Wort ein. Dieser Durchlauf wird so oft wiederholt, bis entweder alle Lücken gefüllt sind, oder in einem Durchlauf kein Wort mehr eingefügt werden konnte. Diese Lücken müssen anhand der Länge dem entsprechenden Wort zugeordnet werden. Dieses Verfahren muss funktionieren, da in jede Lücke nur ein Wort passen kann.

1.2 Umsetzung

2 Dreieckspuzzle

3 Tobis Turnier

4 Streichholzrätsel

Im Folgenden sieht man ein Codebeispiel für Code, der beispielhaft geschrieben wurde.

```
1 // Kommentare sind ne geile Sache
2 import com.badlogic.ShitRenderer;
3
4 public static void main(String[] args){
5     Controller.control(theWorld);
6     Math.atan(1/0);
7 }
```

Im Codebeispiel sieht man, wie der JavaScript Compiler intern arbeitet. Besondere Achtung sollte hierbei dem Math-Befehl gegeben werden, denn Math wurde nicht importiert und deshalb crasht es bereits deshalb, nicht wegen der ZeroDivision, da JavaScript hier einfach das Ergebnis würfeln wurde. # Ehre

5 Wichteln

5.1 Lösungsidee

Im Wesentlichen ist das Problem der Zuordnung mit dem *Stable Marriage Problem* vergleichbar. Es gibt zwei verschiedene Gruppen, die so gut wie möglich verteilt werden müssen. Deshalb lässt sich die Aufgabe mit einer leicht angepassten Variante des *Gale-Shapley* Algorithmus lösen. Allgemein fragen alle Schüler ohne Geschenk bei den

Geschenken an, ob diese noch keinen Partner haben oder den neuen Partner dem aktuellen vorziehen. Dabei wird ein Erstwunsch allem vorgezogen, dann folgen Zweitwunsch, Drittwunsch und zuletzt eine Zuteilung ohne Wunsch. Jeder Schüler fragt nacheinander seine Wünsche an. Sollte er danach noch kein Geschenk haben, versucht er ein übriges Geschenk zu bekommen.

5.2 Umsetzung

Student
hasGift: boolean metWish: int index: int presentId: int wishes: int[] asked: boolean[]
Student(int[], int, int): void requestPresent(Present[], Student[], int, int): void

Abbildung 1: Die Klasse *Student*