

## Programming C++

---

### Project **The Stable–Marriage–Problem**

---

Status:

13. Mai 2022, 17:36

Supervisor: Prof.Dr. G. Haase,

`gundolf.haase@uni-graz.at`

---

#### *The Stable–Marriage–Problem*<sup>1</sup>

Man erzählt sich die folgende Geschichte: Vor vielen Jahren herrschte in einem kleinen Kärntner Bergdorf Aufregung und Unruhe. Zehn junge Frauen hatten die hohe Kunst des Kasnudel-Krendelns erlernt. Sie waren somit im heiratsfähigen Alter und schnellstmöglich zu verheiraten. Wie der Zufall es wollte, gab es in diesem Dorf auch zehn stramme Kandidaten für die Damen. Nun stellte sich die Frage nach der Partnerwahl. Die Dorfgemeinschaft trat zusammen und beschloss, dass Stabilität und Kontinuität essentiell für das Weiterbestehen des Dorfes seien. Deshalb wurde die Organisation der Ehen zur Dorfsache erklärt.

Der weise Dorfälteste wurde mit der Aufgabe betraut, jeder jungen Dame einen passenden Mann auszuwählen. Dieser überlegte, dass stabile Ehen dann erreicht werden können, wenn niemand einen besseren Partner finden kann, der auch gewillt ist, durchzubrennen.<sup>2</sup> Weil der Dorfälteste die Präferenzen aller Kandidaten sehr gut kannte, legte er für jede Frau und für jeden Mann ein Ranking mit den bevorzugten Partnern an, also beispielsweise:

Maria hat am liebsten: 1. Hans; 2. Peter; 3. Josef; 3. Simon; ...

Er formulierte auch einen Algorithmus (heute bekannt als *Gale–Shapley–Algorithmus*<sup>3</sup>), mit dem er vermittels des Rankings stabile Ehen erzeugen konnte:

```
function stableMatching {
    Initialize all m in Men and w in Women to free
    while exists free man m who still has a woman w to propose to {
        w = highest ranked woman to whom m has not yet proposed
        if w is free
            (m, w) become engaged
        else some pair (m', w) already exists
            if w prefers m to m'
                (m, w) become engaged
                m' becomes free
            else
                (m', w) remain engaged
    }
}
```

---

<sup>1</sup><https://youtu.be/Qcv1IqHWAzg>

<sup>2</sup>“Drum prüfe, wer sich ewig bindet, ob sich nicht etwas bessres findet.“ - Verballhornung von Friedrich Schiller, Das Lied von der Glocke.

<sup>3</sup>[http://en.wikipedia.org/wiki/Stable\\_marriage\\_problem](http://en.wikipedia.org/wiki/Stable_marriage_problem)

### Basisaufgabe

Schreiben Sie nun ein Programm, das vermittels einer Präferenzliste stabile Ehen nach dem oben angeführten Algorithmus erzeugt:

- Lesen Sie die jeweiligen Präferenzen der Frauen und Männer aus der Präferenzliste in *preferences.txt*<sup>4</sup> ein. Nutzen Sie für die Speicherung der Präferenzen beispielsweise zwei Objekte `std::map<std::string, std::vector<std::string> >`, wobei das erste Argument den Namen der Frau/des Mannes ist und das zweite Argument ein Vektor mit den Präferenzen.
- Implementieren Sie den *Gale-Shapley-Algorithmus* und speichern Sie generierten Verlobungen in der Map `std::map<std::string, std::string> engaged`. 6 Pkt.
- Legen Sie eine Queue `std::queue<std::string> bachelors` an, die immer die momentan verlobten Männer beinhaltet.
- Geben Sie dabei die Zwischenschritte des Algorithmus' aus, das heißt, wenn Verlobungen generiert werden und diese für eine bessere Paarung wieder aufgelöst werden:  
Maria gets engaged to Hans.  
Maria dumped Hans for Josef.  
...
- Nutzen Sie die STL. Dabei werden `std::map`, `std::vector`, `std::queue` äußerst nützlich sein.
- Haben Sie den Algorithmus korrekt implementiert, erhalten Sie mit der Präferenzliste *preferences.txt* folgende stabile Ehen:  
Esther and Markus; Euphemia and Maxentius; Hannah and Tareq; Jasmin and Rajesh; Lena and Thomas; Ljuba and Leonid; Molly and Mohammad; Nora and Alexeij; Pooja and Daniel; Sara and John
- Hinweis 1: Mittlerweile sind wir im 21. Jahrhundert angekommen. Lässt man auch gleichgeschlechtliche Ehen zu, führt dies zum so genannten *Stable-Roommate-Problem*<sup>5</sup>.
- Hinweis 2: Der Algorithmus generiert stabile Ehen nach der obigen Definition. Das bedeutet aber keineswegs, dass jeder seinen Wunschpartner bekommt.

### Zusatzaufgabe a

Der Gale-Shapley-Algorithmus, so wie er oben formuliert wurde, generiert Ehen, die *male-optimal* sind. Das heißt, für jede andere Paarung bei den Ehen, würde jeder Mann eine Ehefrau bekommen, die weiter hinten auf seiner Präferenzliste steht, oder die gleiche Priorität hat. Frauen werden beim Algorithmus also benachteiligt (= *female-pessimal*). Diese Zusatzaufgabe soll diesen kleinen Schönheitsfehler beheben.

- Analysieren Sie den obigen Algorithmus und stellen Sie fest, warum Männer bevorzugt werden. +1 Pkt.

<sup>4</sup>[http://imsc.uni-graz.at/haasegu/Lectures/Kurs-C/SS21/projects/Stable\\_Marriage\\_Ex/preferences.txt](http://imsc.uni-graz.at/haasegu/Lectures/Kurs-C/SS21/projects/Stable_Marriage_Ex/preferences.txt)

<sup>5</sup>[http://en.wikipedia.org/wiki/Stable\\_roommates\\_problem](http://en.wikipedia.org/wiki/Stable_roommates_problem)

- Modifizieren Sie nun den Algorithmus so, dass er *female-optimal* (und somit *male-pessimal*) ist.
- Implementieren Sie zusätzlich den modifizierten Algorithmus.
- Gibt es Unterschiede bei den generierten Ehen?