

Spieltheorie

B. Nebel, R. Mattmüller
Sommersemester 2012

Universität Freiburg
Institut für Informatik

Übungsblatt P2

Abgabe: Montag, 23. Juli 2012

In diesem Projekt wollen wir uns mit Wahlverfahren und deren Manipulierbarkeit beschäftigen. Ihre Aufgabe wird es sein, ein Programm zu schreiben, das für ein gegebenes Wahlverfahren (Pluralitätswahl, Präferenzwahl mit übertragbaren Stimmen, Borda-Gewinner) sowie für gegebene eigene Präferenzrelation über den Kandidaten und (bekannte) Präferenzrelationen der anderen Wähler über den Kandidaten eine (möglicherweise unwahrhaftige) eigene Präferenzrelation berechnet, die dem Wahlmechanismus gegenüber deklariert wird. Ziel ist es, einen Wahlgewinner zu erzwingen, der in der eigenen wahren Präferenzrelation möglichst weit oben steht. Dabei ist zu beachten, dass die anderen Wähler ihre eigenen Präferenzrelationen gegenüber dem Wahlmechanismus auch nicht wahrhaftig angeben müssen.

Aufgabe P2.1 (Ein- und Ausgabe, 8 Punkte)

Ihr Programm erhält als Eingabe eine Liste aller (wahren) Präferenzrelationen einschließlich der Präferenzrelationen des Wählers, aus dessen Perspektive Sie das Spiel betrachten, die Nummer dieses Wählers, sowie das eingesetzte Wahlverfahren.

Die Liste der Präferenzrelationen wird als Textdatei in diesem Format übergeben:

```
<Anzahl Kandidaten N> <Anzahl Wähler M>
<Präferenzliste von Wähler 0>
<Präferenzliste von Wähler 1>
...
<Präferenzliste von Wähler M-1>
```

Dabei sind die Anzahlen der Kandidaten und Wähler jeweils strikt positive natürliche Zahlen und die Präferenzlisten der einzelnen Wähler leerzeichenge trennte Sequenzen von N paarweise verschiedenen Kandidatennummern zwischen 0 und $N - 1$.

Gibt es etwa zwei Kandidaten k_0 und k_1 und drei Wähler 0, 1 und 2, wobei $k_0 \prec_0 k_1$, $k_0 \prec_1 k_1$ und $k_1 \prec_2 k_0$, so wird das kodiert als

```
2 3
1 0
1 0
0 1
```

Ihr Programm soll wie folgt aufgerufen werden können:

```
$ ./vote <Präferenzrelationen> <Wählernummer> <Wahlverfahren>
```

wobei `<Präferenzrelationen>` der Pfad zu der oben beschriebenen Textdatei mit den Präferenzrelationen ist, `<Wählernummer>` zwischen 0 und $M - 1$ der Index des Wählers, aus dessen Perspektive das Spiel zu betrachten ist, und `<Wahlverfahren>` das Wahlverfahren als einer der drei Strings `plurality`, `irv` oder `borda`.

Die Ausgabe der deklarierten Präferenzliste soll auf der Kommandozeile in dem Format wie in der Eingabe (Leerzeichengetrennte Sequenzen der N paarweise verschiedenen Kandidatennummern zwischen 0 und $N - 1$) erfolgen. Sonst soll nichts ausgegeben werden.

Aufgabe P2.2 (Manipulierte Präferenzrelationen, 3(+3+2) Punkte)

Implementieren Sie für Pluralitätswahl, Präferenzwahl mit übertragbaren Stimmen sowie für die Wahl mit Borda-Gewinner eine geeignete Deklaration der eigenen Präferenzrelation an das Wahlverfahren (Bemerkung: wir nehmen an, dass sich bei allen Wahlverfahren bei Gleichständen immer der Kandidat mit niedrigerem Index durchsetzt). Beachten Sie, dass es hier immer eine triviale Lösung gibt, nämlich die wahrheitsgemäße Deklaration der eigenen Präferenzrelation.

Um die Aufgabe etwas interessanter zu gestalten, veranstalten wir einen Wettbewerb zwischen allen eingereichten Programmen, bei dem das Programm gewinnt, das am erfolgreichsten darin ist, bevorzugte Kandidaten des von ihm vertretenen Wählers zum Wahlgewinner zu machen. Dazu generieren wir Listen von zufälligen Präferenzrelationen und verteilen sie unter den Teilnehmern. Für jede solche „Wahl“ erhält jeder Spieler (jedes Programm) Punkte, die anschließend aufsummiert werden. Die Punkte für eine einzelne „Wahl“ nach Wahlverfahren f und ein Programm, das Wähler i mit Präferenzrelation \prec_i vertritt, ergeben sich wie folgt: Seien $\prec'_0, \dots, \prec'_{M-1}$ die von den Wählern deklarierten Präferenzrelationen und $k = f(\prec'_0, \dots, \prec'_{M-1})$ der gewählte Kandidat. Dann erhält das Programm, das Wähler i vertritt, p Punkte, wobei p die Position des gewählten Kandidaten k in der wahren Präferenzrelation \prec_i von Wähler i ist (je höher, umso besser).

Die Punkte für diese Übungsaufgabe verteilen sich wie folgt: je ein Punkt für geeignete Präferenzdeklarationen für die drei Wahlverfahren, je ein Punkt für das beste Programm pro Wahlverfahren, und zwei Punkte für das Programm mit den meisten Punkten insgesamt.

Die Projekte dürfen und sollten in Gruppen von zwei Studenten bearbeitet werden. Sie dürfen dazu die Programmiersprachen C, C++, Java, Python und Haskell verwenden. Bitte fügen sie außerdem eine README-Datei mit einer Anleitung zum Kompilieren und Ausführen an. Schicken sie ihre Programme und Ergebnisse an engesset@informatik.uni-freiburg.de.