

## Homework #1

<b>Deadline: Tuesday, 5 October 2010, 11:00</b>
---

### Question 1 (10 marks)

In analogy to the definition of Condorcet winners, a *Condorcet loser* is a candidate that would lose against any other candidate in a pairwise contest.

- Give an example that shows that the plurality rule *can* elect a Condorcet loser.
- Prove that the Borda rule *never* elects a Condorcet loser.

*Remark:* It is in fact possible to show that the Borda rule is the *only* positional scoring rule (with a strictly descending scoring vector) that satisfies this property.

### Question 2 (10 marks)

In the late 1980s, Hervé Moulin published a paper showing that every voting procedure that satisfies the Condorcet Principle allows for situations where a voter has an incentive to not vote at all rather than to vote sincerely. Track down the paper in question, give a precise statement of the result mentioned above, and write up a succinct proof, using the notation and terminology from the course slides where applicable.

### Question 3 (10 marks)

Analyse the computational complexity of the winner determination problem for the system of Single Transferable Vote (STV).

### Bonus Question (20 marks)

In 1876, Charles Lutwidge Dodgson (better known as Lewis Carroll, the author of *Alice in Wonderland*) proposed the following voting procedure. Each voter strictly ranks all alternatives. The *score* of candidate  $x$  is the minimal number of *swaps* of adjacent alternatives in a voter's ranking required to make  $x$  a Condorcet winner. The alternative(s) with the lowest score win(s). Implement the Dodgson rule. You may use any programming language you wish; the only constraint is that your program must run on the standard Linux environment provided by the FNWI. Your solution should be well documented: describe your algorithm and how you addressed the challenge of implementing a computationally intractable rule; provide comments on the code itself; clearly explain how to run your program; provide some examples that can be used for testing; and include an evaluation of the performance of your program. Your program should take as input a text file using a syntax as in the following example (which represents a ballot profile for three voters and four alternatives):

```
a1 > a3 > a2 > a4
a1 > a4 > a3 > a2
a2 > a3 > a1 > a4
```

*Note:* I will accept solutions for this question until 1 December 2010.