

Algorithmique et programmation fonctionnelle

UE INF 201,231,251

PROJET – COMMENT VOTER ?

Pierre Wulles

2023

Consignes :

- Le langage de programmation utilisé sera OCaml
- Vous pouvez (et devez quand c'est nécessaire !) commenter votre code en le mettant entre `(*...*)`
- Les programmes devront être écrits en utilisant la partie d'OCaml enseignée dans cette UE, c'est à dire en particulier sans boucles `for` `while`, ni référence et tableaux.
- Dans toute question vous pouvez utiliser une fonction d'une question précédente et ce même si vous n'avez pas su écrire la fonction
- Vous ne pouvez utiliser une fonction auxiliaire que si vous la définissez et la réalisez vous même avant

Le but de ce sujet est de comparer plusieurs méthodes de scrutins. Expliquer les avantages et inconvénients de chacun.

1 Introduction

La majorité des démocraties dans le monde a recours au scrutin uninominal à un ou deux tours pour élire ses représentants. Or ce mode de scrutin peut poser problèmes et paradoxes, voici quelques exemples :

- **Présidentielles de 1988** : François Mitterand gagne contre Jacques Chirac mais aurait perdu contre Raymond Barre qui n'arrive pas au second tour.
- **Présidentielles de 1995** : Jacques Chirac gagne contre Lionel Jospin au second tour, mais si Philippe de Villiers n'était pas candidat, Edouard Balladur aurait pu devancer Chirac au premier tour.
- **Présidentielles US de 2000** : G.W. Bush est élu mais si Ralph Nader ne s'était pas présenté, Al Gore aurait été élu.
- **Présidentielles de 2002** : Lionel Jospin était donné gagnant contre Jean-Marie Le Pen et contre Jacques Chirac au second tour mais c'est Jacques Chirac qui a été élu face à Jean-Marie Le Pen. Vraisemblablement, sans les candidatures de Christiane Taubira et Jacques Chevènement, Lionel Jospin aurait été au second tour. Jacques Chirac avait lui-même soutenu la candidature de Christiane Taubira pourtant dans le camp opposé. Charles Pasqua avait renoncé à se présenter au dernier moment de peur de priver la droite d'un second tour.
- **Présidentielles de 2007** : Nicolas Sarkozy est élu mais selon tous les sondages François Bayrou était donné gagnant contre tous les candidats au second tour.
- **Présidentielles de 2017** : Sans la candidature de Benoît Hamon, Jean-Luc Mélenchon pouvait arriver au second tour.

Ces exemples le montrent, le modèle traditionnel ne semble pas traduire la réelle volonté du peuple car le gagnant dépend en partie du jeu des candidatures. En effet l'apparition de « petits candidats » peut faire perdre un camp par rapport à un autre et la stratégie dite de « vote utile » peut pousser les électeurs à voter pour un candidat dont ils ne veulent pas réellement en pensant qu'il a plus de chance qu'un autre. Le vote blanc n'est pas non plus reconnu par ces systèmes. Le but de ce sujet est d'explorer d'autres méthodes de scrutin qui ont pour objectif de résoudre les problèmes précédemment mentionnés.

2 Scrutin uninominal

Attention

Dans cette partie toutes les fonctions utilisées doivent être spécifiés et réalisés. En particulier on n'utilisera aucune fonction issue des modules de la librairie standard d'OCaml.

Un bulletin sera représenté par une chaîne de caractères (qui représente un candidat) et une urne sera représentée par une liste de bulletin. Le score d'un candidat est le nombre de votants pour ce candidat. Le panel est la liste des candidats. On supposera que l'on a jamais d'ex-aequo entre deux candidats car en pratique cela est très improbable.

Question 1 Proposer des types OCaml pour représenter un candidat, un bulletin, une urne, le score d'un candidat et un panel de candidats.

On se place pour le moment dans le cas d'un scrutin uninominal à un tour.

Pour l'exemple considérons que l'on a trois candidats : Keny, Kyle et Stan et 16 électeurs. L'urne et le panel ressemblent à ceci :

```
let lc1 = ["Eric"; "Kyle"; "Stan"];;  
let u1 = ["Eric"; "Kyle"; "Stan"; "Kyle"; "Kyle"; "Stan"; "Eric"; "Eric"; "Kyle"; "Eric"; "Stan"; "Eric";  
         "Eric"; "Eric"; "Stan"; "Stan"];;
```

Question 2 Écrire une fonction OCaml `compte c u` qui pour une urne donnée renvoie le score du candidat `c`. Par exemple :

```
val compte : candidat -> urne -> score = <fun>  
# compte "Eric" u1;;  
- : score = 7
```

Question 3 Écrire une fonction `depouiller lc u` qui en prenant le panel et l'urne renvoie la liste des `candidat*score`, on pourra introduire un type `resultat` pour représenter cette liste. **Tous les candidats** du panel doivent apparaître dans le résultat, même si ils n'obtiennent aucune voix, **l'ordre** dans lequel ils apparaissent doit **être celui du panel**.

```
val depouiller : panel -> urne -> resultat list = <fun>  
# depouiller lc1 u1;;  
- : (candidat * score) list = [("Eric", 7); ("Kyle", 4); ("Stan", 5)]
```

Question 4 Écrire une fonction `union r1 r2` qui fait l'union des résultats. Cela pourra par exemple représenter la mise en commun des résultats de deux bureaux de votes.

```
val union : resultat -> resultat -> resultat = <fun>
```

Question 5 Écrire une fonction `max_depouille l` qui en prenant la liste des `candidat*score` renvoie le candidat qui a le meilleur score, et son score.

```
val max_depouiller : resultat -> candidat * score = <fun>
```

Question 6 En utilisant les fonctions précédentes, écrire une fonction `vainqueur_scrutin_uninominal u lc` qui pour une urne et un panel donné renvoie le vainqueur de l'élection, c'est à dire le candidat ayant obtenu le plus de voix. On supposera qu'il n'y a pas d'égalité possible.

```
val vainqueur_scrutin_uninominal : urne -> panel -> candidat = <fun>
```

Question 7 En pratique pour donner plus de poids au résultat d'un scrutin uninominal on veut s'assurer que le candidat élu recueille au moins la moitié des suffrages. Pour cela on organise un **second tour** avec les deux premiers candidats du premier tour. C'est ce système qui est utilisé en France pour les élections présidentielles ou législatives. Écrire une fonction `deux_premiers` qui à partir du dépouillage d'une urne extrait les deux premiers. On pourra pour cela s'aider d'une fonction `suppr_eleme l e` qui supprime une occurrence de l'élément `e` dans la liste `l`.

```
val suppr_elem : 'a list -> 'a -> 'a list = <fun>  
val deux_premiers : urne -> panel -> (candidat * score) * (candidat * score) = <fun>
```

Question 8 Dans cette question, nous allons au travers d'une étude de cas, mettre en évidence un problème de ce mode de scrutin. On suppose maintenant que, toutes choses égales par ailleurs, un quatrième candidat fait son apparition, en conséquence certains électeurs reportent leurs votes sur ce candidat :



FIGURE 1 – Philippe Geluck

```
let lc2 = ["Eric"; "Kyle"; "Stan"; "Kenya"];
let u2 = ["Kenya"; "Kyle"; "Kenya"; "Kyle"; "Kyle"; "Kenya"; "Eric"; "Eric"; "Kyle"; "Eric"; "Stan"; "Eric";
"Eric"; "Eric"; "Stan"; "Stan"];
```

Quel est le problème? Ce phénomène se nomme **paradoxe d'Arrow**. Identifier les situations similaires dans l'introduction et trouver un exemple de paradoxe d'Arrow dans le paysage politique français récent, par exemple dans l'élection présidentielle de 2022.



FIGURE 2 – Stan, Kyle, Eric et Kenny (Southpark)

Question 9 Identifier précisément la source du problème dans le scrutin **uninominal**.

3 Jugement majoritaire

4 Reconstitons les voix

5 Conclusion

Comme on a pu le voir il est compliqué de retranscrire la volonté du peuple au travers des bulletins de vote. Les deux méthodes présentées ici en tant qu'alternative au scrutin uninominal à deux tours possèdent. Il existe aussi d'autres méthodes pour choisir un candidat qui repose sur le hasard comme le **scrutin de Condorcet randomisé** mais elles sont très mal acceptées par la population car non déterministes. Tirer au sort un candidat est une méthode qui a également ses partisans et présente de nombreux avantages, elle était d'ailleurs utilisée à l'époque de la démocratie athénienne.

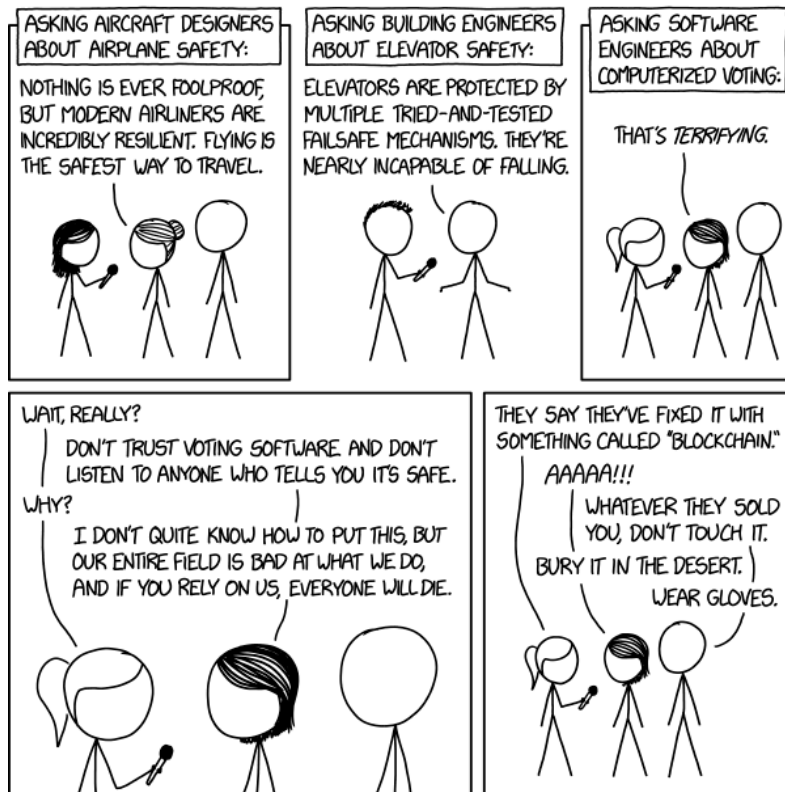


FIGURE 3 –

Question 10 Conclure en comparant les trois méthodes de votes présentées dans ce devoir. Identifier au moins un avantage du scrutin uninominal par rapport aux deux autres modes de scrutin présentés. Faire des recherches pour montrer en quoi le paysage politique français aurait pu être modifié si on avait eu recours à d'autres modes de scrutin.

Références

- Majority Judgment, Measuring, Ranking and Electing, *Michel Balinski and Rida Laraki*, 2010
- Sur la forme des élections, *Nicolas De Condorcet*