

RAPPORT ANALYSE SYSTEME DE VOTE

- AMETTLER Thibault
- BEDOS Enzo
- MAOULIDA ATTOUMANI Yazid-Raoul
- COMBELONGE Enzo
- LIMOSANI Francesco

<https://github.com/TibOwO/SAE-1.01>

Dans cette SAE, nous devons arriver à simuler l'élection du meilleur jeu avec 4 systèmes de votes.

Dans ce rapport, Nous verrons en détails comment fonctionne chaque système de votes en passant en revue leurs avantages et inconvénients. Nous verrons ensuite la structure du code ainsi que les fichiers d'entrée. Enfin, nous verrons le fichier de sortie et les fichier Oracle.

Comment fonctionne chaque système de votes

Vote Pluralité

Le vote pluralité est le système de vote le plus facile à élaborer et à mettre en œuvre. Son fonctionnement est le suivant :

- 1 Tour
- 1 Vote par votant
- Le candidat avec le + de voix gagne l'élection

Ce système est fréquemment utilisé notamment au premier tour des élections présidentielle en France.

Son principal avantage réside dans sa facilité de mise en œuvre et de comptage.

Son principal inconvénient est son manque de précision pour représenter l'opinion publique. En effet, les votants doivent choisir un candidat malgré la possibilité que plusieurs puissent leur correspondre ou alors qu'aucun ne leur corresponde.

En ce sens, Le Vote pluralité ne permet pas de sonder une population de manière optimisée mais sa facilité de mise en œuvre lui permet d'être l'un des systèmes de votes les plus utilisés aujourd'hui.

Vote Négatif

Le vote négatif est un système de vote permettant d'exprimer une voix contre un candidat. Selon ce principe, L'élus sera donc pas le meilleur mais le moins pire d'entre tous, un autre débat s'ouvre alors, Le moins pire est-il synonyme de meilleur, nous préférons laisser le lecteur en décider.

En attendant, ce système reposant sur le même fonctionnement que le Vote Pluralité, il suffit, en théorie, d'inverser l'ordre des gagnants du vote pluralité pour obtenir ceux du Vote négatif.

C'est d'ailleurs sur ce concept que nous avons basé l'élaboration du code du Vote Négatif.

Vote Préférentiel

Le Vote préférentiel est un système permettant d'exprimer une préférence pour tous les candidats.

Ce système permet à contrario du Vote Pluralité de représenter bien plus précisément l'opinion d'une population. En effet Les votants peuvent, grâce à ce système exprimer bien plus clairement leurs idées en les classant par ordre de préférence. Voici son fonctionnement :

- 4 votes OBLIGATOIRE par votants
- Le premier dans la liste de préférence rapporte 4 points au candidat
- Le dernier dans la liste de préférence rapporte 1 point au candidat
- Le candidat avec le + de points gagne

Vote Cumulatif

Le vote cumulatif permet, à la manière du vote préférentiel, au votant de mieux exprimer ses préférences sur les différents candidats. Le vote cumulatif permet au votant de mieux exprimer les nuances de son vote comparé au vote préférentiel. Son fonctionnement est le suivant :

- x Points par votants
- Chaque votant peut choisir de donner ses points à n'importe lequel des candidats
- Il peut choisir plusieurs candidats (min: 1 max: 4)

Structure du Code/Fichier d'entrée

Le programme doit être lancé avec le terminal grâce à la commande suivante :

```
./vote <Input.txt> sortie.txt
```

Le fichier d'entrée

Le fichier d'entrée est constitué de 2 parties :

- Les jeux
- Les votes

JEUX

Les jeux doivent être déclarer au tout début du fichier d'entrée sous cette forme :

«Jeu 1» «Jeu 2» «Jeu 3» «Jeu 4»

Ex :

```
1 MK CSGO SF CVL
```

LES VOTES

Cumulatif

Les votes sont pris en compte grâce à un mot : « vote : ». Lorsque le programme va rencontrer ce mot, Il va comptabiliser les 4 prochains mots comme des votes.

Les votes sont attribués aux différents candidats suivant l'ordre d'entrée des jeux expliqué ci-dessus.

```
1 Jeu1 Jeu2 Jeu3 Jeu4
2
3 NOM
4 Prénom
5 vote: 90 8 1 1
```

Ici, Jeu1 se verra attribuer 90 points

Jeu2 8 points

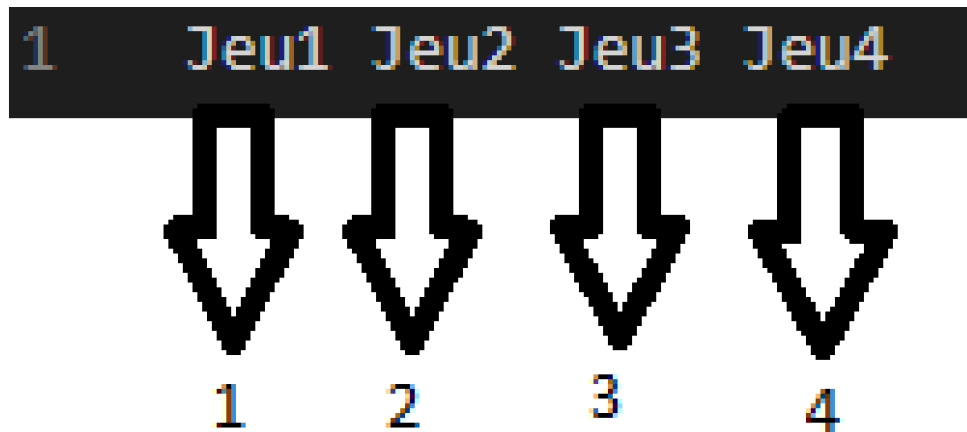
Jeu3 et Jeu4 seulement 1 point

Préférentiel

A la manière du vote cumulatif ci-dessus, le vote préférentiel prendra en compte les votes lorsqu'il lira le mot « vote : »

A partir de là, suivant le vote du votant, le jeu se verras attribuer un nombre de points allant de 4 points (pour le premier de la liste) à 1 point (pour le dernier de la liste)

Ici les nombres correspondent à l'ordre d'entrée des jeux lors de leur déclaration au début du fichier (le premier jeu de la liste correspondant au numéro 1 le deuxième au numéro 2 etc...)



Voici un exemple de ce à quoi pourrait ressembler un vote :

```
1  Jeu1 Jeu2 Jeu3 Jeu4
2
3  Nom
4  Prénom
5  vote: 4 1 2 3
```

Ici, le jeu 4 prendrais 4 points

Le jeu 1 prendrais 3 points

Le jeu 2 prendrais 2 points

Le jeu 3 prendrais 1 points

Pluralité/Négatif

Le fichier d'entrée pour le système de vote Pluralité ou Négatif se tiennent sous la même forme, seul le classement final change.

Le programme prendra en compte un vote que lorsqu'un chiffre entre 1 et 4 sera lu.

Le système attribue donc 1 point au candidat respectif.

```
3    NOM
4    Prénom
5    3
```

Ici, Le jeu 3 sera donc incrémenté / décrémenté de 1.

Structure du code

Le code est découpé en 5 procédures et 1 fonction :

- AfficheJeux () : Définit les 4 premiers mots du fichier d'entrée comme étant les 4 jeux à départager.
- Comptevote () : Comptabilise les voix de chaque votant et les enregistre dans les variables correspondantes.
 - C'est cette procédure qui est changée selon le système de vote utilisé.
- AfficheResultat () : Affiche votes obtenu pour chaque jeu.
- CalculGagnant (jeu1, jeu2, jeu3, jeu4) : Tri les votes comptabilisés pour les classer dans l'ordre et déterminer le gagnant.
- AffichGagnant () : Affiche le résultat en les classant du premier au dernier.

Fichier de sortie

Notre programme une fois exécuté, va écrire toutes les informations dans un fichier de sortie

Ce fichier peut changer légèrement selon le vote utilisé car par exemple le vote cumulatif exprime le résultat en pourcentage

(« sortie.txt »). Ce fichier sera créé s'il n'existe pas.

Voici une représentation du fichier sortie une fois le votePluralité exécuté.

```
1  Les quatre jeu sont :
2  Mariokart
3  StreetFighter
4  CournterStrike
5  Civilisation
6  Resultat final :
7
8  Mariokart = 24
9  StreetFighter = 35
10 CournterStrike = 28
11 Civilisation = 29
12 Nombre de participant = 113
13
14 1er, StreetFighter avec un score de : 35
15 2ème, Civilisation avec un score de : 29
16 3ème, CournterStrike avec un score de : 28
17 4ème, Mariokart avec un score de : 24
18
```

Fichier Oracle

Le fichier oracle est un fichier permettant de vérifier la sortie du programme. Celui-ci est composé de la sortie théorique du programme. Il est sensé reproduire au caractère près le fichier sortie.

Pour vérifier que le programme est « juste », il suffit de faire la différence entre le fichier oracle et le fichier sortie.