Algorithms and Data Structures

Set, Map and Stream processing for Elections Results

Assignment-4

Version: February 10th, 2023

Introduction

At https://data.overheid.nl/datasets you can find various public data sets that are published by the Dutch government. Specifically, at https://data.overheid.nl/dataset/verkiezingsuitslag-tweede-kamer-2021 you find all results of the Dutch 2021 national elections for our parliament.

That gives you the opportunity to perform some independent quality checking on the published data and do a full recount of those election results as you practice your programming skills with Java Sets, Maps and Streams!

After completing reading of this document, you can start to prepare your solution with help of the provided starter project. It is your job to complete the missing code in the provided classes. These sections are marked with // TODO comment lines. You can verify the basics of your solution with the provided unit tests. We also expect you to add additional test classes with extra unit tests and prepare a report with code explanations and output results.

Understanding the source data

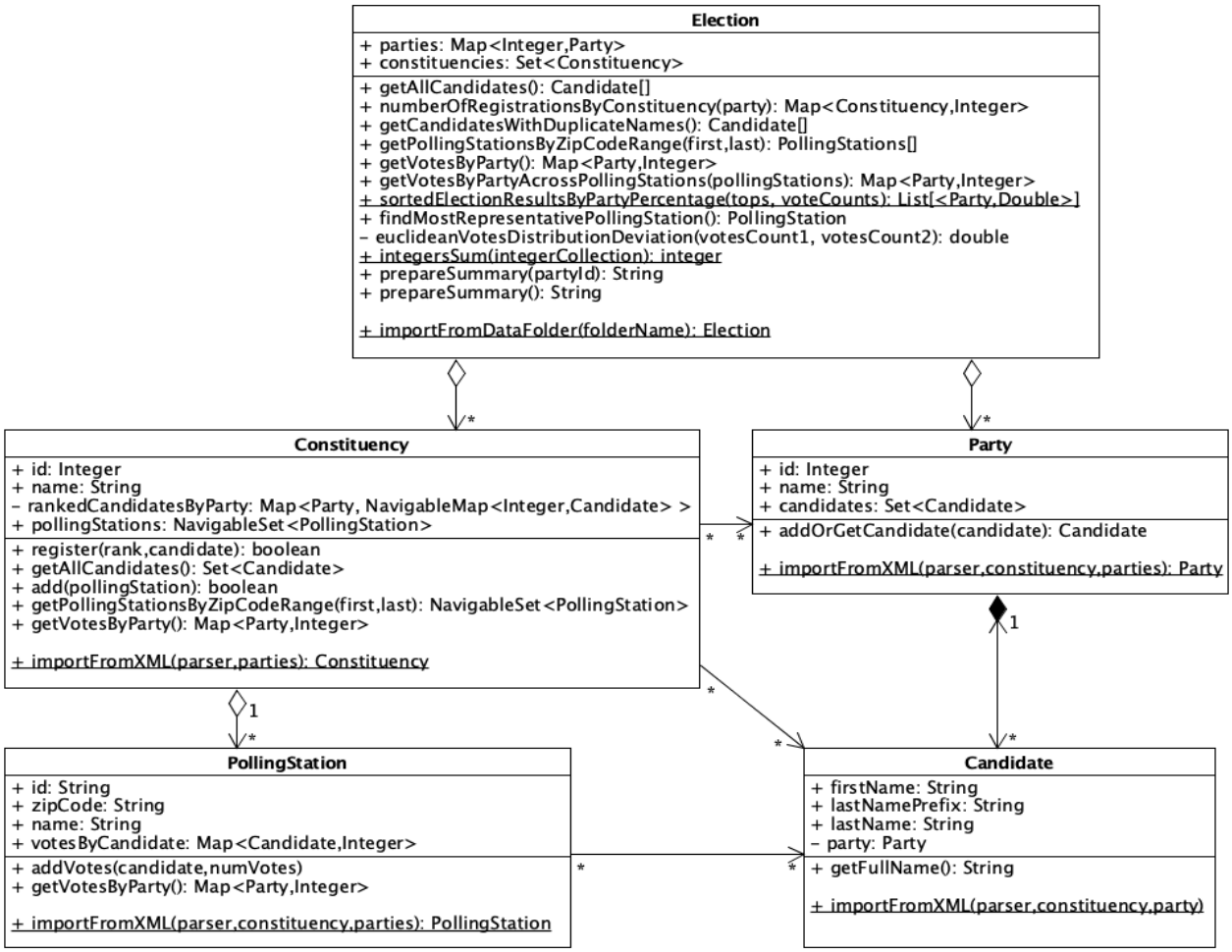
The Dutch government supports Open Data initiatives and aspires to use open data standards for sharing data sets. At https://www.kiesraad.nl/verkiezingen/osv-en-eml/eml-standaard you can read how they apply the OASIS 'Election Markup Language' (EML) for distributing elections results. This XML based format is rather extensive, but in the starter project of this assignment we've already provided the methods to load these types of data sets and you do not have to worry about that.

The total data set of the 2021 national elections is quite large: about 2Gb. So, it has not been included into the starter project. You should download that data yourselves, in three parts from https://data.overheid.nl/dataset/verkiezingsuitslag-tweede-kamer-2021 (Select the tab 'databronnen'; download EML_bestanden_TK2021_deel_[123].zip). Once you've unpacked the .zips, you should merge the contents into a single folder for combined processing.

However, this dataset is not suitable for efficient testing after initial development, because it may take up to a minute to fully load the data into the memory of your computer. Therefore, a small extract 'EML_bestanden_TK2021_HvA_UvA' has been provided in the resources folder of the starter project. That set is used by all unit tests in the starter project. When you pass all unit tests and your other validations, only then you should use the full size data set to produce the outcome of your solution…

Understanding the class hierarchy

The starter project model the Dutch election system with five classes, primarily. These classes are explained here, and further modeled in below class diagram. This model is a simplification of the full capabilities and content of the EML files: We only discuss what we need for the purpose of this assignment.



- A **Candidate** represents a person who participates in the elections via membership of a **Party**. All candidates shall have a unique full name (firstName + lastNamePrefix + lastName) within their Party, but we do find duplicate full names of Candidates of different parties.
- A **Party** is an organization of Candidates with a shared vision about how our country should be run. The total number of votes won by all Candidates in a Party will determine how many seats that Party will be awarded in the parliament.
  Each Party is uniquely identified by a sequence number (party-id) on the national ballot list.
  In the full data set you'll find that 37 different Parties have participated in the 2021 elections.
- Parties shall register their Candidates at Constituencies (kieskringen). A **Constituency** represents a geographical area of the nation (e.g. a district or province). The Netherlands has been subdivided into 20 Constituencies. Every citizen will be invited to vote within his or her Constituency of residence.
  Parties may choose to participate in only few Constituencies and Parties may also register a different list of Candidates in each Constituency. On the ballot list, Candidates will be ranked within the Party into a specified order and that order may also vary across constituencies.
  For that, each Constituency class tracks a map of ranked Candidate lists by Party.
- The actual voting occurs in a **PollingStation** (stembureau). Each Constituency manages several PollingStations and each PollingStation tracks the number of votes that have been cast on each Candidate. Every PollingStation in the same Constituency uses the same ballot list of candidates of that Constituency.
  It was the intention of the EML designer that Polling Stations are uniquely defined by an Id. But in practice three municipalities have separately registered vote counts of different days or votes

by mail into different entries of Polling Stations with duplicate Ids…. @@**$!#. In this assignment, we identify Polling Stations uniquely by zip code and id, and the vote counts of duplicate entries will be merged by the data loader…

Objectives

The main objectives of Assignment-4 are:

O1. Validate integrity of the data set by cross-calculating numbers of different candidates, registered candidates, and casted votes overall, by party, by constituency or by subset of polling stations.
O2. Reproduce the overall national election result in percentages by Party.
O3. Find the most representative polling station, which is the polling station that has the most similar election result when compared with the overall national outcome.

This last objective to find 'the most similar election result' is not yet defined SMARTly here... Data scientists have proposed multiple 'similarity measures' which can be used to quantify similarity of datasets. At https://towardsdatascience.com/9-distance-measures-in-data-science-918109d069fa you can read a.o. about Euclidian, Manhattan, and Chebyshev distance metrics. Below you find an example of how the Euclidian distance of two vote distributions is calculated (as the sum of the squares of the differences with the national distribution):

	Party A	Party B	Party C	***Euclidian distance***
National votes distribution:	0.50	0.30	0.20	
Polling Station X:	0.40	0.40	0.20	
difference:	*-0.10*	*+0.10*	*0*	***0.0200***
Polling Station Y:	0.35	0.35	0.30	
difference:	*-0.15*	*+0.05*	*+0.10*	***0.0350***

Here the votes distribution of Polling Station X is rated closer to the National average than the votes distribution of Polling Station Y.

The starter project already provides a method to calculate the Euclidian distance between two votes count results. You can use this method to find the polling station with the minimum distance to the national votes count.

Approach

We recommend the following approach to complete this assignment:

1. Unpack the starter project into your git repository and commit and push your baseline.
2. Complete the // TODO items in the Candidate and Party classes and verify associated unit tests.
3. Complete the // TODO items in the PollingStation and Constituency classes and verify unit tests.
4. Complete the // TODO items in the Election class while progressing with associated unit tests. The main class uses the Election.prepareSummary methods to produce the final report.
5. Verify the outcome of the main program against the console output at the end of this document.
6. Leverage the capabilities of Sets, Maps and Streams in your implementations. Specifically, you shall (investigate and) use each of the following Java methods at least once in a purposeful way:
    a. entrySet(), keyset() or values()
    b. .subSet() or submap()
    c. .map() or .mapToInt() or .mapToDouble() or .mapToObj()
    d. .flatMap()

e. .filter() or limit() or .sorted()
f. .max() or .min() or .count() or .sum() or .average() or .reduce() or findFirst()
g. .get() or .orElse()
h. .collect() or .toList() with Collectors.toMap() or Collectors.groupingBy()
i. Comparator.comparing() or Map.Entry.comparingByValue() or .comparingByKey()

7. Download the full data set of the election results from https://data.overheid.nl/dataset/verkiezingsuitslag-tweede-kamer-2021 (Select the tab 'databronnen') and Run the main program against that full data and include the console output in your report. (The full dataset shall **NOT BE ADDED TO YOUR GIT**. It will exceed storage capacity limit of your repository and block further access. Use a different data folder in your local file system, and update the main program with that folder path)
8. Prepare your report as indicated below.

Other tips and constraints.

- Signatures of public methods shall not be changed.
  You may add private methods and instance variables as you deem appropriate.
- Apply proper encapsulation to all your coding.
  Isolate duplicate or similar functionality into reusable methods.
  Reuse existing methods as much as possible.
- Minimize cyclomatic and cognitive complexity.
  Add useful comment lines.
  Use meaningful variable and method names.
  Derived data that is used multiple times within a single method is calculated only once.
- Your code should pass all unit tests with green ticks.

Unit tests

1) Some unit tests are provided to assist you to verify correctness of your code.
   These should all pass with green ticks.
2) But these unit tests are not exhaustive for all possible scenarios:
   Add at least two additional unit tests in a separate test class for scenario's that were impacted by a defects in your initial development work but were not easy traceable from the available unit tests.
3) Add another unit test class that verifies outcome of at least three specific queries within the prepareSummary methods.

Report

Your report shall include the following:

1. Seven code snippets with explanation/justification of your code, with clear rationale.
   (A Dutch or English transcription of the statements in your code does not add any information.)
   E.g.: explain the content of a stream after every intermediate step in plain language.
2. Explanation of differences in your console output of the main program when compared with the given output at the end of this document (even when all your unit tests are green.)
3. Console output of running your solution against the **full data set**, as downloaded earlier.

Grading

Grading criteria are according to guidance in the study manual and as per rubric at the DLO.

Expected console output from 'EML_bestanden_TK2021_HvA_UvA'

```
Election summary of
/Users/somej/GitRepositories/2223/ADS/assignments/A4_Elections_solution/target/classes/EML_bestanden_TK2021_HvA_UvA:


36 Participating parties:

[Party{id=1,name='VVD'}, Party{id=2,name='PVV (Partij voor de Vrijheid)'}, Party{id=3,name='CDA'}, Party{id=4,name='D66'},
Party{id=5,name='GROENLINKS'}, Party{id=6,name='SP (Socialistische Partij)'}, Party{id=7,name='Partij van de Arbeid
(P.v.d.A.)'}, Party{id=8,name='ChristenUnie'}, Party{id=9,name='Partij voor de Dieren'}, Party{id=10,name='50PLUS'},
Party{id=11,name='Staatkundig Gereformeerde Partij (SGP)'}, Party{id=12,name='DENK'}, Party{id=13,name='Forum voor
Democratie'}, Party{id=14,name='BIJ1'}, Party{id=15,name='JA21'}, Party{id=16,name='CODE ORANJE'},
Party{id=17,name='Volt'}, Party{id=18,name='NIDA'}, Party{id=19,name='Piratenpartij'}, Party{id=20,name='LP (Libertaire
Partij)'}, Party{id=21,name='JONG'}, Party{id=22,name='Splinter'}, Party{id=23,name='BBB'}, Party{id=24,name='NLBeter'},
Party{id=25,name='Lijst Henk Krol'}, Party{id=26,name='OPRECHT'}, Party{id=27,name='JEZUS LEEFT'}, Party{id=28,name='Trots
op Nederland (TROTS)'}, Party{id=29,name='U-Buntu Connected Front'}, Party{id=30,name=''}, Party{id=31,name='Partij van de
Eenheid'}, Party{id=32,name='DE FEESTPARTIJ (DFP)'}, Party{id=33,name='Vrij en Sociaal Nederland'}, Party{id=34,name='Wij
zijn Nederland'}, Party{id=36,name='De Groenen'}, Party{id=37,name='Partij voor de Republiek'}]

Total number of constituencies = 2

Total number of polling stations = 4

Total number of candidates in the election = 1.119

Different candidates with duplicate names across different parties are:

[Candidate{partyId=30,name='Felix Tangelder'}, Candidate{partyId=33,name='Felix Tangelder'},
Candidate{partyId=30,name='Christian Kromme'}, Candidate{partyId=33,name='Christian Kromme'},
Candidate{partyId=30,name='Theo Vos'}, Candidate{partyId=33,name='Theo Vos'}]


Overall election results by party percentage:

[Party{id=4,name='D66'}=26.902173913043477, Party{id=5,name='GROENLINKS'}=13.768115942028986,
Party{id=17,name='Volt'}=9.782608695652174, Party{id=1,name='VVD'}=9.646739130434783, Party{id=9,name='Partij voor de
Dieren'}=8.695652173913043, Party{id=14,name='BIJ1'}=7.653985507246377, Party{id=7,name='Partij van de Arbeid
(P.v.d.A.)'}=6.25, Party{id=12,name='DENK'}=3.9855072463768115, Party{id=6,name='SP (Socialistische Partij)'}=3.125,
Party{id=2,name='PVV (Partij voor de Vrijheid)'}=1.9927536231884058, Party{id=13,name='Forum voor
Democratie'}=1.9474637681159421, Party{id=3,name='CDA'}=1.6757246376811594, Party{id=18,name='NIDA'}=1.4945652173913044,
Party{id=15,name='JA21'}=0.9510869565217391, Party{id=8,name='ChristenUnie'}=0.36231884057971014,
Party{id=19,name='Piratenpartij'}=0.3170289855072464, Party{id=22,name='Splinter'}=0.2717391304347826,
Party{id=10,name='50PLUS'}=0.22644927536231885, Party{id=20,name='LP (Libertaire Partij)'}=0.22644927536231885,
Party{id=11,name='Staatkundig Gereformeerde Partij (SGP)'}=0.18115942028985507, Party{id=16,name='CODE
ORANJE'}=0.18115942028985507, Party{id=24,name='NLBeter'}=0.1358695652173913,
Party{id=21,name='JONG'}=0.04528985507246377, Party{id=23,name='BBB'}=0.04528985507246377, Party{id=25,name='Lijst Henk
Krol'}=0.04528985507246377, Party{id=27,name='JEZUS LEEFT'}=0.04528985507246377, Party{id=31,name='Partij van de
Eenheid'}=0.04528985507246377, Party{id=26,name='OPRECHT'}=0.0, Party{id=28,name='Trots op Nederland (TROTS)'}=0.0,
Party{id=29,name='U-Buntu Connected Front'}=0.0, Party{id=33,name='Vrij en Sociaal Nederland'}=0.0, Party{id=36,name='De
Groenen'}=0.0, Party{id=37,name='Partij voor de Republiek'}=0.0]


Polling stations in Amsterdam Wibautstraat area with zip codes 1091AA-1091ZZ:

[PollingStation{id='0363::SB126',zipCode='1091GH',name='Stembureau Hogeschool van Amsterdam, Wibauthuis'},
PollingStation{id='0363::SB159',zipCode='1091GH',name='Stembureau Hogeschool van Amsterdam, Leeuwenburg'}]

Top 10 election results by party percentage in Amsterdam area with zip codes 1091AA-1091ZZ:

[Party{id=4,name='D66'}=27.083333333333332, Party{id=5,name='GROENLINKS'}=11.155913978494624,
Party{id=1,name='VVD'}=9.879032258064516, Party{id=17,name='Volt'}=9.744623655913978, Party{id=9,name='Partij voor de
Dieren'}=8.736559139784946, Party{id=7,name='Partij van de Arbeid (P.v.d.A.)'}=6.720430107526882,
Party{id=14,name='BIJ1'}=6.115591397849462, Party{id=12,name='DENK'}=5.309139784946237, Party{id=6,name='SP
(Socialistische Partij)'}=3.8306451612903225, Party{id=2,name='PVV (Partij voor de Vrijheid)'}=2.4193548387096775]

Most representative polling station is:

PollingStation{id='0363::SB126',zipCode='1091GH',name='Stembureau Hogeschool van Amsterdam, Wibauthuis'}

[Party{id=4,name='D66'}=26.957494407158837, Party{id=5,name='GROENLINKS'}=11.856823266219239,
Party{id=1,name='VVD'}=9.395973154362416, Party{id=17,name='Volt'}=9.284116331096197, Party{id=7,name='Partij van de
Arbeid (P.v.d.A.)'}=8.501118568232663, Party{id=9,name='Partij voor de Dieren'}=8.389261744966444,
Party{id=14,name='BIJ1'}=6.263982102908278, Party{id=12,name='DENK'}=4.586129753914989, Party{id=6,name='SP
(Socialistische Partij)'}=4.026845637583893, Party{id=13,name='Forum voor Democratie'}=2.237136465324385,
Party{id=18,name='NIDA'}=2.0134228187919465, Party{id=3,name='CDA'}=1.5659955257270695, Party{id=2,name='PVV (Partij voor
de Vrijheid)'}=1.45413870246085, Party{id=15,name='JA21'}=1.1185682326621924,
Party{id=19,name='Piratenpartij'}=0.5592841163310962, Party{id=10,name='50PLUS'}=0.33557046979865773, Party{id=20,name='LP
(Libertaire Partij)'}=0.33557046979865773, Party{id=8,name='ChristenUnie'}=0.22371364653243847,
Party{id=11,name='Staatkundig Gereformeerde Partij (SGP)'}=0.22371364653243847, Party{id=16,name='CODE
ORANJE'}=0.22371364653243847]


Summary of Party{id=17,name='Volt'}:

Total number of candidates = 28

Candidates: [Candidate{partyId=17,name='Ernst Boutkan'}, Candidate{partyId=17,name='Bibi Wielinga'},
Candidate{partyId=17,name='Elske Kroesen'}, Candidate{partyId=17,name='Marieke Koekkoek'},
Candidate{partyId=17,name='Sarah de Koff'}, Candidate{partyId=17,name='Itay Garmy'}, Candidate{partyId=17,name='Jeroen van
Iterson'}, Candidate{partyId=17,name='Sacha ten Hove'}, Candidate{partyId=17,name='Martin Gravelotte'},
Candidate{partyId=17,name='Frank Toeset'}, Candidate{partyId=17,name='Robine van Eck'}, Candidate{partyId=17,name='Sylvia
van Laar'}, Candidate{partyId=17,name='Ger van Eeden'}, Candidate{partyId=17,name='Laurens Dassen'},
Candidate{partyId=17,name='Jeroen Koendjbiharie'}, Candidate{partyId=17,name='Joris van Oppenraaij'},
Candidate{partyId=17,name='Nilüfer Gündoğan'}, Candidate{partyId=17,name='Reinier van Lanschot'},
Candidate{partyId=17,name='Theo Doreleijers'}, Candidate{partyId=17,name='Ilca Italianer'},
Candidate{partyId=17,name='Floris Eigenhuis'}, Candidate{partyId=17,name='Michelle van Zanten'},
Candidate{partyId=17,name='Friso Datema'}, Candidate{partyId=17,name='Fons Janssen'}, Candidate{partyId=17,name='Marleen
Ramaker'}, Candidate{partyId=17,name='Thomas van der Meer'}, Candidate{partyId=17,name='Katya Lenskaya'},
Candidate{partyId=17,name='Sandra Griffejoen'}]
```

```
Total number of registrations = 56
Number of registrations per constituency: {Constituency{id=9,name='Amsterdam'}=28, Constituency{id=11,name='Den
Helder'}=28}
```