<div align="center">

School of Computing & Information Technology

CSCI251/CSCI851 Advanced Programming Autumn 2020

</div>

Assignment 2 (Worth 1% (Part A) + 14% (Part B))

Due 11:55pm Monday 1st June 2020 (Start of Week 12).
Early diagram submission: 7pm Monday 11th May 2020 (Start of Week 9).

Overview

This assignment is to be implemented using object oriented programming. It involves implementing a simulation of an election campaign and the election itself.

In addition to providing code you need to do the following:

- Submit a draft UML like class diagram indicating how your classes are structured and related.

- Submit a final report, which contains a final UML like class diagram reflecting the structure in your code, and text/tables/diagrams addressing the points below:

  1. Describe qualitatively, so not numerically, the five (5) different issues that you have chosen.

  2. Describe the three (3) political parties and specify the range of stances on each of the issues.

  3. Describe the characteristics of your electorates, and how the stance distribution for the electorates is modelled.

  4. The winner for an electorate is the candidate for that electorate who obtains the most votes. You need to describe how the number of votes for each candidate is obtained. There should be randomness in this process. While the process doesn't need to be particularly complicated, there should be dependence on the relative stances of electorates and candidates, and on candidate popularity.

  5. Describe the characteristics of the candidates and the qualitative impact they have.

  6. Describe the characteristics of the other people associated with the political parties and the qualitative impact they have.

  7. Describe what local and national events you have chosen, and how the event mechanisms qualitatively depend on the characteristics of the components described above.

General notes

These are some general rules about what you should and shouldn't do.

1. Your assignment should be sensibly organised with the same kind of expectations in this area as assignment one, although you may reasonably have more files for this assignment. No memory leaks etc. too.

2. **Other than the initial command line input, the program should run without user input. In particular this means there shouldn't be pauses waiting for the user to press a key.**

3. **There shouldn't be pauses generally, please don't use sleep!**

4. Your code must compile on capa with the compilation instructions you provide in `Readme.txt`. If it doesn't you will likely receive zero for the coding part of the assignment.

5. You have to use classes, and should make use of encapsulation.

6. You must use inheritance. If you don't use inheritance you will lose a fair few marks.

7. You can use polymorphism, if it's appropriate.

8. If you use operator overloading, it should be appropriate.

9. You can use your own data files to import for names or similar information.

10. Output should only be to standard out or standard error.

1 A political episode

The nation is on edge ahead of the impending election campaign. The leaders have presented their candidates and published manifestos detailing their plans for the nation. You need to design and build a simulation of the campaign and subsequent election.

Your code should compile on capa according to instructions you provide in a `Readme.txt` file.

Once your program is compiled into the executable `APE`, it must run as follows:

./APE n m

where n is the number of electorates in the nation, and m is the number of campaigning days prior to the election. The value of n should be in the range 1 to 10 inclusive. The value of m should be in the range 1 to 30 inclusive.

On running, `APE` should report on each party, so their stance values and the relevant details of their candidates, leader, and other significant figures.

`APE` should also report on the state of the nation, so give details on the electorates including the stance distribution.

2   Components

You need to give consistent and appropriate definitions for a class hierarchy associated with the various components of the campaign and election.

2.1   Issues

You need to decide on five (5) issues of relevance, so concerns, for the nation. These can be real–world national issues, such as global warming, real–world local issues, such as road infrastructure, or not–so–likely issues, such as the distribution of cat food. For the purpose of simplicity we model a stance on an issue as being two–dimensional, with the dimensions being:

1. The significance of the issue.

2. The approach taken to the issue. You don't need to specify what the approach means, it's just a scale where similar values mean similar approaches and quite different values mean quite different approaches.

Each party, candidate, and electorate will have different stances on each. Stance values may be changed by events.

2.2   Political parties and candidates

Each political party has the following officials:

1. One leader, not themselves a candidate for an electorate.

2. One candidate for each electorate.

3. One campaign manager for each electorate.

4. One national campaign manager.

5. One national financial manager.

Each political party has a range across each stance. You define this for each of your three political parties. Initial stance values should be randomly generated as somewhere in that range, both for the party, party candidates, and the party leader.

Both the leader and the candidates will have popularity, and at least one other characteristic. It's up to you to determine how those are relevant but both popularity and the other characteristic(s) must have some impact.

2.3   Electorates

You need to determine appropriate components to represent the electorates. Each will have a political stance distribution rather than a specific stance. It's up to you to decide how this works. If you choose to trivialise it to being a single stance, as for the candidates and parties, you won't get full marks for this part.

2.4 Campaign days and events

Each day of the campaign involves the candidates and leaders attempting to convince the electorates to support them.

Each campaign will receive funding based on the effectiveness of their financial manager and the support for the party. That funding will be spent on the campaign and the impact should be some function of the campaign managers. Campaign managers should determine how much to spend under the assumption that they aim to use all their money by the last day of campaigning, with the restriction that they cannot spend money on the day it comes in.

On every day of the campaign, for each electorate, there is a 20% chance of a local event:

1. Local debate.

2. Candidate related event. You need two different types of these events.

3. Local issue impact event. You need two different types of these events.

Local events only have an impact within an electorate. Candidates and campaign managers will have some impact on the outcomes.

On every day of the campaign there is a 30% chance of a national event:

1. Leaders debate.

2. Leader related event. You need two different types of these events.

3. National issue impact event. You need two different types of these events.

National events should have some impact across all electorates. Leaders and national campaign managers will have some impact on the outcomes.

It is up to you to determine the likelihood of individual events but they should be weighted towards less significant events. You should describe the events and their impact in your report.

Each campaign day `APE` should report the day, what has happened in each electorate, and the significant impacts that have occurred. `APE` should also report the finances of each party at the end of each day, so initial, income, expenditure, and final.

2.5 Election day and wrapping up...

Immediately prior to election day you should report on each party, so their stance values and the relevant details of their candidates, leader, and other significant figures.

You also need to report on the state of the nation, so give details on the electorates including population and stance distribution.

For each candidate you need to provide the breakdown of votes and identify the winner, so the candidate with the most votes.

You then need to provide a national summary, so how many electorates have been won by each party. If a party has won more than 50% of the electorates, they will form a government and you can report the party leader as the new leader for the country. If no party has won more than 50% of the electorates, `APE` should declare a hung parliament.

Notes on submission

Submission is via Moodle.

Your code must compile on capa with the instructions you provide. If it doesn't you will likely be given zero for the programming part of this assignment.

Please submit your source, so .cpp and .h files, Readme.txt file, any data files used, report, and makefile if you have one, in a zip file `A2.zip`. There shouldn't be any directory structure within the zip file.

Make sure your report is in pdf and has your name, student number, and lab class marked clearly on it. The final diagram should be in your final report pdf and not in a distinct document. It may be ignored if it's a non–pdf format.

Your early diagram submission should also be in pdf, and should also have your name, student number, and lab class on it. You can split it across multiple diagrams if necessary.

1. The early diagram submission is **7pm Monday 11th May 2020 (Start of Week 9).**

2. The main submission deadline is **11:55pm Monday 1st June 2020 (Start of Week 12).**

3. Late submissions will be marked with a 25% deduction for each day, including days over the weekend.

4. Submissions more than three days late will not receive marks, unless an extension has been granted.

5. If you need an extension apply through SOLS, if possible **before** the assignment deadline.

6. Academic misconduct is treated seriously. Students involved will likely receive zero.