**Lab 3 - Voting System**

**Introduction**

This lab is about designing an object-oriented voting system, written in java, that will illustrate the principles of object-oriented programming.

**Classes**

**Person (Person.java)**

This will be an abstract class. In this class we have used the concept of inheritance as the candidate class and voter class are child classes so they can access the methods of parent class.

Must have the following variables:

● age : int - private

● gender : char - private, will hold values like M, F, T, U

● firstName : String - protected

● lastName : String - protected

● politicalParty : String - protected, will hold values like Democrat, Republican, Non-Affiliate

Must have the defined constructor:

● Person ( )

● Person (int age, char gender, String firstName, String lastName, String politicalParty) Which will set the passed properties to our defined variables

Will implement the following public methods:

● getAge() - returns int

● getGender() - returns char

Will define the following abstract methods:

● getFullName() - returns String

**Candidate (Candidate.java)**

This class holds the information about the candidate it will extend the class Person.

Must have the following variables:

● voteCount : int - private, when the class is instantiated this should default to 0

Must have the defined constructor:

● Candidate ( )

● Candidate (int age, char gender, String firstName, String lastName, String politicalParty) Which will set the passed properties to our defined variables and super class

Will implement the following public methods:

● getVoteCount() - returns int

● incrementVoteCount() - void, increments the voteCount

● getFullName() - returns String

○ If the candidates firstName = “John” and lastName = “Smith” and political party is “Democrat” then it returns “John Smith - D”

○ If the candidates firstName = “John” and lastName = “Smith” and political party is “Republican” then it returns “John Smith - R”

○ If the candidates firstName = “John” and lastName = “Smith” and political party is “Non-Affiliate” then it returns “John Smith - NA”

○ Otherwise “John Smith”

**Voter (Voter.java)**

This class will gather the information about the voter it will extend the class Person.

Must have the following variables:

● voterId : int - private

● voted : boolean - private, when the class is instantiated this should default to false

Must have the defined constructor:

● Voter ( )

● Voter (int voterId, int age, char gender, String firstName, String lastName, String politicalParty)

Which will set the passed properties to our defined variables and super class

Will implement the following public methods:

● getVoterId() - returns int

● hasVoted() - returns boolean

● voted() - void, sets the voted flag to true

● getFullName() - returns string - e.g. firstName = “John” and lastName = “Smith” then it returns “John Smith”

**Voting Machine (VotingMachine.java)**

This class is used to manipulate and print the results (i.e. who is the winner and how many votes he/she gained). This class has crucial role in whole system

Must have the following variables:

● candidates : Candidate[ ] - public, array of all candidates in the race

Must have the defined constructor:

● VotingMachine (Candidate candidates[ ]) Which will set the passed properties to our defined variables

Will implement the following public methods:

● vote(Voter v, Candidate c) - void, counts vote (increments candidate count) and marks voter as voted

○ We will not handle the scenario that a voter votes for a candidate not in the election

● vote(Voter v) - void, if voter has not voted, marks voter as voted.

● tally() - void, prints results and winner.

For example we have the following candidates

(Sam Smith - R, Joe Dean - D, Jane Doe - NA) the out print will be like

Sam Smith - R has 5 votes.

Joe Dean - D has 2 votes.

Jane Doe - NA has 7 votes.

Jane Doe - NA won with 7 votes.

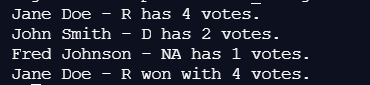
We will not handle the case for ties at the moment, so the first candidate to have the

most votes wins.

**Testing**

You can test your codes by compile and execute the provided file-Election.java.

You will see the printout as



**Submission**

After completing this lab, zip up all the .java files in the name of the

zip file must be COP3809\_Lab3\_SYY\_XXXX.zip; where YY is 01, 02 or 03 and XXXX

is your student Id.