HY-150 2023

4η Σειρά Ασκήσεων - Live

05/05/2023

18:00 – 20:00

Καλείστε να υλοποιήσετε χρησιμοποιώντας τους κανόνες της κληρονομικότητας ένα πρόγραμμα στο οποίο οι χρήστες, θα πρέπει να αποφασίσουν ποιο κόμμα και ποιο υποψήφιο να ψηφίσουν.

1. ***Party Classes (50%)***

Θα πρέπει μέσω της κλάσης PoliticalParty να υλοποιήσετε 2 υπο-κλάσεις; BlueParty, και το RedParty. Η κάθε μια από αυτές είναι ένα διαφορετικό εκλογικό κόμμα. Δίνονται επίσης δύο βοηθητικά structs τα Info και Nominee, στα οποίο θα αποθηκεύονται οι πληροφορίες του κάθε κόμματος και του κάθε υποψήφιου. Κάθε υπο-κλάση θα πρέπει μέσο του constructor της, να εκχωρεί τιμές στις μεταβλητές **partyInfo**, **Name**, **Nominees**.

Η κλάση PoliticalParty περιέχει 2 virtual συναρτήσεις (**GetTotalScore, GetNomineesVotes)** οι οποίες λειτουργούν διαφορετικά σε κάθε υπο-κλάση.

- Για το BlueParty το **GetTotalScore** υπολογίζετε ως ο συνολικός αριθμός ψήφων δια τον συνολικό αριθμό των υποψήφιων. Ενώ για το RedParty υπολογίζετε ως ο συνολικός αριθμός ψήφων.
- Για το BlueParty το **GetNomineesVotes** επιστρέφει, όλους τους υποψήφιους, και τον αριθμό των ψήφων που έχουνε ταξινομημένα με αύξουσα σειρά ως προς τον όνομα των υποψήφιων ενώ για το RedParty το **GetNomineesVotes** επιστρέφει, την ίδια πληροφορία ταξινομημένη ως προς τις ψήφους, του κάθε υποψήφιου.

```cpp
class PoliticalParty{
    protected:
        Info partyInfo;
        string Name;
        vector<Nominee> Nominees;

    public:
        virtual int GetTotalScore() =0;
        virtual vector<string> GetNomineesVotes() = 0;

        Info GetPartInfo(){
            return partyInfo;
        }

        void SetPartyInfo(Info partyInfo){
            this->partyInfo = partyInfo;
        }

        void SetNominees(vector<Nominee> noms){
            Nominees = noms;
        }

        void IncreaseVote(int idx){
            Nominees[idx].Votes++;
        }

        string GetAllInfo(){
            //Add code here
        }

};
```

```cpp
struct Info{
    int foundedYear;
    string chairmanName;
    string Ideology;
};

struct Nominee{
    string Name;
    int Votes;
};
```

2. ***Init & Run (50%)***

Το πρόγραμμα αρχικά θα πρέπει να φτιάχνει ένα object τύπου RedParty και ένα object BlueParty. Στην συνέχεια θα διαβάζει από τα αρχεία RedParty.txt και BlueParty.txt την τωρινή κατάσταση των υποψήφιων και θα αποθηκεύει την πληροφορία στα vectors Nominees. Στην συνέχεια θα ζητάει από τον χρήστη να επιλέξει ένα κόμμα. Στην συνέχεια θα του εκτυπώνει όλη την πληροφορία του κόμματος χρησιμοποιώντας την συνάρτηση **GetAllInfo()**. Τέλος, ο χρήστης θα διαλέγει έναν από τους υποψήφιους. Το πρόγραμμα τελειώνει όταν ο χρήστης δώσει την τιμή -1.

Η συνάρτηση GetAllInfo() θα πρέπει να επιστρέφει ένα string με την εξής μορφή:

Political Party: RedParty
Founded: 1955
ChairmanName: Giorgos Spirakis
Ideology: We like red.
Score: 15
Nominees:
[0]: Papadimitropoulos Michalis 12
[1]: Lefterakias Lefteris 8
….

Οδηγίες

Παραδώστε όλα τα αρχεία κώδικα (.h, .cpp) που φτιάξατε για την άσκηση και προαιρετικά τα CMakelists.txt, όλα μέσω του elearn submit, όπως και όλες τις προηγούμενες ασκήσεις έως τώρα. **Είναι υποχρεωτικό στις κλάσεις τα declarations και definitions να βρίσκονται σε χωριστά αρχεία.**