

ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

OOP- Master Chef Part D

Το παιχνίδι

Η φετινή εργασία αποτελεί μια απλουστευμένη παραλλαγή του δημοφιλούς παιχνιδιού Master Chef (περισσότερες λεπτομέρειες στο <http://www.star.gr/tv/masterchef/>). Ο σκοπός είναι να χτίσουμε σιγά σιγά ένα πρόγραμμα που θα εξομοιώνει το παιχνίδι με έναν όσο το δυνατόν πιο ρεαλιστικό τρόπο (Εικόνα 1).



Εικόνα 1: Master Chef 2018

Στη τέταρτη εργασία, το ζητούμενο είναι να υλοποιήσετε τις κλάσεις Voting και Votes που θα χρησιμοποιηθούν για την προσομοίωση της ψηφοφορίας που λαμβάνει χώρα στην Ομάδα που έχασε την Ομαδική Δοκιμασία. Πιο συγκεκριμένα, θα πρέπει να χρησιμοποιήσετε κάποιες δυναμικές δομές που βρίσκονται στην Standard Template Library της C++ για να δημιουργήσετε τις κατάλληλες συναρτήσεις.

ΠΡΟΣΟΧΗ!!! Σε καμία περίπτωση δεν πρέπει να χρησιμοποιήσετε τον κώδικα από την προηγούμενη εργασία σας για να αντικαταστήσετε αυτόν που δίνεται ως έτοιμος για την τέταρτη εργασία.

Περιγραφή του Παιχνιδιού

Για τις ανάγκες της τέταρτης εργασίας, ο κώδικας που θα σας δοθεί περιλαμβάνει μόνο το κομμάτι του προγράμματος που χρειάζεται για την υλοποίηση της ψηφοφορίας. Πιο συγκεκριμένα, η διαδικασία έχει ως εξής:

Το παιχνίδι βρίσκεται αρχικά σε μια τυχαία μέρα του Master Chef πριν από την πρώτη αποχώρηση. Οι δύο Ομάδες είναι πλήρεις από άποψη παικτών, η κούραση, η τεχνική κατάρτιση και η πείνα κάθε παίκτη έχουν τυχαίες τιμές.

Αυτό που θα προσομοιώνεται στο παρόν κομμάτι της εργασίας είναι η μέρα Ψηφοφορίας. Όταν εκκινεί η μέρα Ψηφοφορίας, ο χρήστης επιλέγει ποια από τις δύο Ομάδες έχασε στον Ομαδικό Διαγωνισμό και θα περάσει από το Αγώνισμα της Ασυλίας.

Η διαδικασία του Αγωνίσματος Ασυλίας είναι ήδη υλοποιημένη και μετά το πέρας της ένας παίκτης της Ομάδας έχει κερδίσει την Ασυλία και έχει και περισσότερες ψήφους από τους υπόλοιπους (2 ή 3).

Σειρά έχει η διαδικασία της Ψηφοφορίας για την Ομάδα που έχασε, καθώς και η Αποχώρηση ενός παίκτη, τις οποίες καλείστε να υλοποιήσετε.

Εργασία Δ' – Υλοποίηση Δυναμικών Δομών Δεδομένων

Το τέταρτο σκέλος της εργασίας απαιτεί την δημιουργία της κλάσης **Voting**, καθώς και της βοηθητικής κλάσης **Vote**. Ακολουθεί αναλυτική παρουσίαση των κλάσεων και των μελών τους που καλείστε να υλοποιήσετε.

Υλοποίηση Κλάσεων

1. Κλάση Vote (Vote.h):

Η κλάση αυτή δημιουργεί τα αντικείμενα τύπου Ψήφος. Περιλαμβάνει τις εξής μεταβλητές:

1. **voted**: είναι ένα αλφαριθμητικό που περιλαμβάνει το όνομα του Παίκτη που ψηφίστηκε για αποχώρηση.
2. **reason**: είναι ένα αλφαριθμητικό το οποίο περιέχει τον λόγο αποχώρησης.

Οι συναρτήσεις που πρέπει να υλοποιηθούν για αυτή τη κλάση είναι **τουλάχιστον** οι εξής:

- a. **Συναρτήσεις Αρχικών Συνθηκών**: θα πρέπει να υλοποιηθούν 2 συναρτήσεις Αρχικών Συνθηκών (**μια χωρίς ορίσματα και μια που να δέχεται για ορίσματα τον Παίκτη που ψηφίστηκε για αποχώρηση και τον λόγο ψήφου, με αυτή τη σειρά**), οι οποίες έχουν ως στόχο την αρχικοποίηση όλων των μεταβλητών που περιλαμβάνει η κλάση. Τα ορίσματα θα πρέπει να τα αντιστοιχίσετε στις κατάλληλες μεταβλητές της κλάσης.
- b. Όλες οι συναρτήσεις **get** και **set** για όλες τις μεταβλητές της κλάσης.
- c. Μια συνάρτηση **status** που να εκτυπώνει τις τιμές των μεταβλητών σε μια δεδομένη στιγμή.

2. Κλάση Voting (Voting.h και Voting.cpp):

Η κλάση αυτή **ΔΕΝ** θα δημιουργεί αντικείμενα/στιγμιότυπα, καθώς όλα τα χαρακτηριστικά της θα είναι **static**. Αυτό σημαίνει ότι όλες οι κλήσεις θα γίνονται με το όνομα της κλάσης και τον τελεστή επίλυσης εμβέλειας. Θα περιλαμβάνει τις εξής μεταβλητές:

1. **votes**: είναι ένα **διάνυσμα (δυναμικός πίνακας)** αντικειμένων τύπου **Votes**, που θα περιλαμβάνει **όλες τις ψήφους** που κατατέθηκαν κατά την διάρκεια του Συμβουλίου.
2. **results**: είναι ένας **χάρτης (δυναμική δομή)** η οποία θα περιλαμβάνει τα αποτελέσματα της ψηφοφορίας. Τα **κλειδιά** του χάρτη θα είναι τα ονόματα των παικτών που ψηφίστηκαν (τύπου **string**) και **οι τιμές** θα είναι οι ψήφοι που συγκέντρωσε ο καθένας (τύπου **int**).

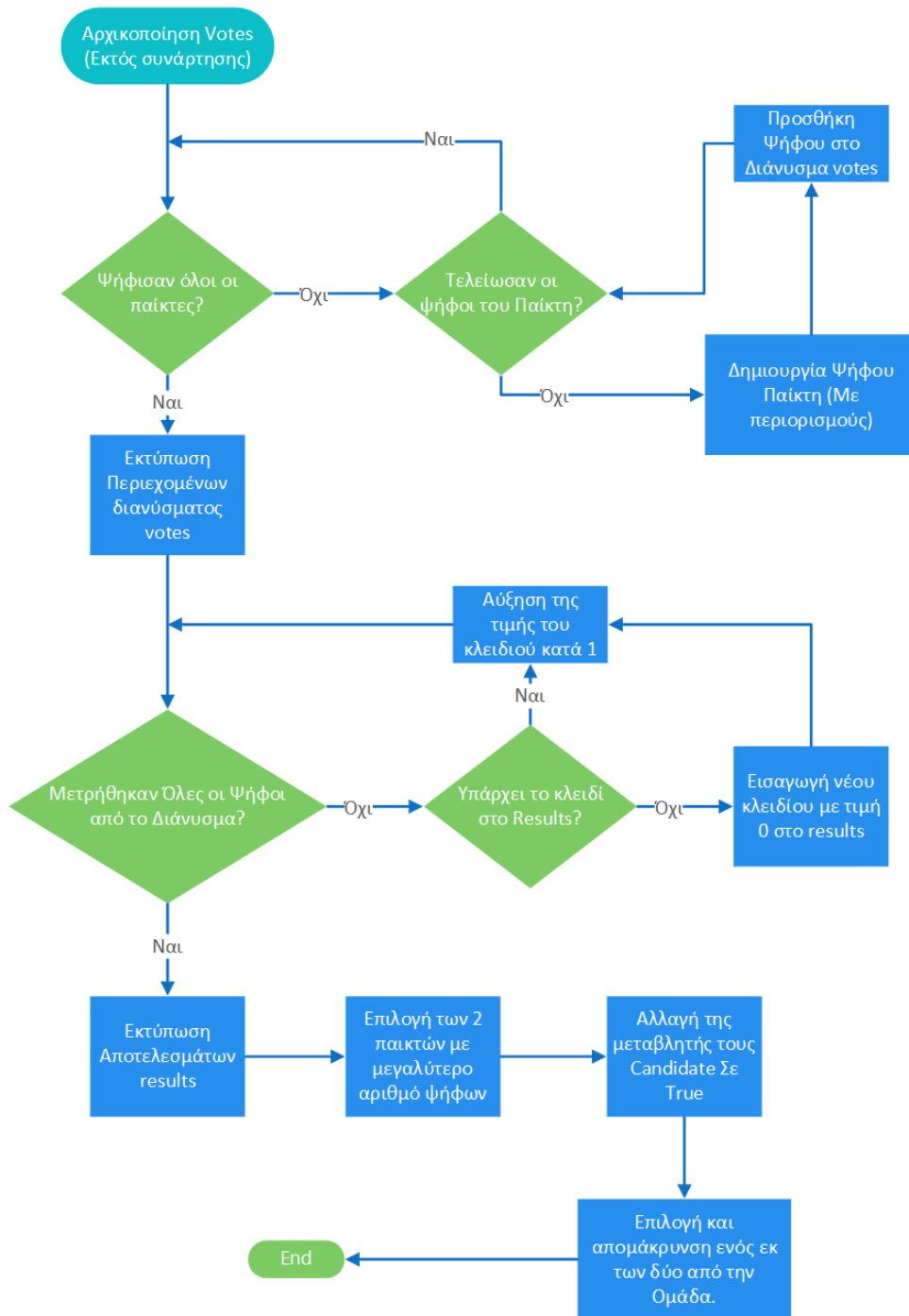
Οι συναρτήσεις που πρέπει να υλοποιηθούν για αυτή τη κλάση είναι **τουλάχιστον** οι εξής:

- a. Μια συνάρτηση **static void votingProcess(Team &team)** η οποία θα περιλαμβάνει ολόκληρη τη διαδικασία της Ψηφοφορίας: την **κατάθεση των ψήφων**, την **καταμέτρηση** και την **εξαγωγή των αποτελεσμάτων**. Στην **συνέχεια** θα πρέπει να **επιλέγονται οι 2 παίκτες που έχουν τους περισσότερους ψήφους** και να **αποχωρεί ο παίκτης με την μικρότερη τεχνική κατάρτιση**. Τη συνάρτηση αυτή μπορείτε (και προτείνεται) να την σπάσετε σε μικρότερα κομμάτια, κατά βούληση.

Προσοχή!!!

1. Κατά την διάρκεια της ψηφοφορίας, ένας παίκτης **δεν μπορεί να ψηφίσει τον εαυτό του, κάποιον παίκτη που έχει αποχωρήσει και κάποιον παίκτη που έχει ασυλία**.
2. **Ο παίκτης που έχει κερδίσει ασυλία έχει περισσότερες από μια ψήφους που πρέπει να χρησιμοποιήσει.**
3. Ο τρόπος με τον οποίον θα ψηφίζουν οι παίκτες **αφήνεται στην ευχέρειά σας**. Μπορείτε να κάνετε ότι θέλετε, από τυχαίες επιλογές μέχρι στρατηγική ψήφιση του καλύτερου παίκτη ή του παίκτη με τις περισσότερες νίκες.
4. Αν κατά την καταμέτρηση, 2 ή περισσότεροι παίκτες έχουν τον ίδιο αριθμό ψήφων, τότε **η ισοπαλία θα πρέπει να επιλυθεί με κάποιον τρόπο από εσάς** (κρατάτε τον πρώτο, διαλέγετε στην τύχη κάποιον, κτλ.)

Voting Process



Εικόνα 2 Ψευδοκώδικας Υλοποίησης της votingProcess της Voting.

Βοηθητικές Κλάσεις

Παρακάτω, παρουσιάζονται όλες οι βοηθητικές κλάσεις, με τις μεταβλητές/συναρτήσεις που **χρειάζεστε** για να υλοποιήσετε σωστά τις ζητούμενες κλάσεις. Αυτές έχουν ήδη υλοποιηθεί από εμάς και μπορείτε να τις χρησιμοποιήσετε ως έχουν.

ΠΡΟΣΟΧΗ!!! Σε καμία περίπτωση δεν επιτρέπεται να επέμβετε στον κώδικα των κλάσεων και των συναρτήσεων αυτών. Σε περίπτωση που το κάνετε, η εργασία σας αυτομάτως θεωρείτε λανθασμένη και μηδενίζεται.

1. Κλάση Team (Team.h και Team.cpp):

Η κλάση αυτή περιλαμβάνει όλες τις μεταβλητές και τις συναρτήσεις που χρειάζονται για την υλοποίηση του αντικειμένου της Ομάδας. Οι μεταβλητές της κλάσης Ομάδα που σας ενδιαφέρουν είναι οι εξής:

1. **string color**: το αλφαριθμητικό που συμβολίζει το χρώμα της Ομάδας.
2. **Player *players**: πίνακας αντικειμένων τύπου **Player** που περιλαμβάνει τους παίκτες της Ομάδας.

Για κάθε μια από τις παραπάνω μεταβλητές ορίζονται οι αντίστοιχοι **setters** και **getters**, τους οποίους μπορείτε να τους χρησιμοποιήσετε κατά βούληση. Εκτός αυτών, οι συναρτήσεις που μπορείτε να χρησιμοποιήσετε είναι οι εξής:

1. **int getNumberOfPlayers()**: Συνάρτηση που επιστρέφει **τον αριθμό των παικτών** σε μια Ομάδα.
2. **void setCandidate(string name)**: Συνάρτηση που αλλάζει την μεταβλητή candidate ενός παίκτη σε **true**, χρησιμοποιώντας το όνομα του παίκτη ως είσοδο.
3. **void removePlayer(string name)**: Η συνάρτηση αυτή **διαγράφει** έναν συγκεκριμένο παίκτη από την Ομάδα, χρησιμοποιώντας το όνομα του παίκτη ως είσοδο.
4. **void status()**: Συνάρτηση που να εκτυπώνει τις τιμές των μεταβλητών της κλάσης εκείνη τη στιγμή.

2. Κλάση Player (Player.h και Player.cpp):

Η κλάση αυτή περιλαμβάνει όλες τις μεταβλητές και τις συναρτήσεις που χρειάζονται για την υλοποίηση του αντικειμένου του Παίκτη. Οι μεταβλητές της κλάσης Παίκτη που σας ενδιαφέρουν είναι οι εξής:

1. **string name:** το αλφαριθμητικό που συμβολίζει το όνομα του Παίκτη.
2. **float technique:** ένας δεκαδικός αριθμός που συμβολίζει την **τεχνική ικανότητα του Παίκτη** σε ποσοστό %.
3. **float fatigue:** ένας δεκαδικός αριθμός που συμβολίζει την **Κούραση του Παίκτη** σε ποσοστό %.
4. **float popularity:** ένας δεκαδικός αριθμός που συμβολίζει την **δημοφιλία του Παίκτη** σε ποσοστό %.
5. **int wins:** ένας ακέραιος αριθμός που συμβολίζει τον **αριθμό των νικών** που έχει κάνει ο παίκτης στα αγωνίσματα.
6. **int votes:** ένας ακέραιος αριθμός που δείχνει τον **αριθμό των φωνών που μπορεί να ψηφίσει ο συγκεκριμένος παίκτης**.
7. **bool immunity:** μια μεταβλητή που δείχνει αν ο συγκεκριμένος παίκτης **έχει ασυλία ή όχι**.
8. **bool candidate:** μια μεταβλητή που δείχνει αν ο συγκεκριμένος παίκτης **είναι υποψήφιος για αποχώρηση ή όχι**.

Για κάθε μια από τις παραπάνω μεταβλητές ορίζονται οι αντίστοιχοι **setters και getters**, τους οποίους μπορείτε να τους χρησιμοποιήσετε κατά βούληση. Εκτός αυτών, οι συναρτήσεις που μπορείτε να χρησιμοποιήσετε είναι οι εξής:

1. **void status():** Συνάρτηση που να εκτυπώνει τις τιμές των μεταβλητών εκείνη τη στιγμή.

3. Αρχείο main.cpp:

Το αρχείο αυτό, περιλαμβάνει και κάποιες καθολικές μεταβλητές που πρέπει να χρησιμοποιήσετε για να υλοποιήσετε το πρόγραμμά σας σωστά. Αυτές είναι οι εξής:

1. **Team teams[]:** είναι ο πίνακας που περιέχει τις δύο ομάδες που διαγωνίζονται. Στην **θέση 0** είναι η **Κόκκινη Ομάδα**, ενώ στην **θέση 1** είναι η **Μπλε Ομάδα**. Τα στοιχεία των Ομάδων και των Παικτών είναι συμπληρωμένα ήδη από εμάς.

Εγκατάσταση

Για να εγκαταστήσετε το Project στο CodeBlocks, αρκεί να αποσυμπιέσετε το αρχείο SurvivorPartD.rar σε μια τοποθεσία του σκληρού σας δίσκου που έχετε τα προγράμματά σας, και στη συνέχεια από το περιβάλλον του CodeBlocks να κάνετε:

1. Import New Project (από την αρχική σελίδα που φαίνεται στο CodeBlocks)
2. Επιλέξτε το MasterChef.cpb (Project File) από τον φάκελο που μόλις αποσυμπιέσατε το Project.

Οδηγίες

Τα προγράμματα θα πρέπει να υλοποιηθούν σε C++, με πλήρη τεκμηρίωση του κώδικα. Κάθε αρχείο στο πρόγραμμά σας θα πρέπει να περιέχει επικεφαλίδα σε μορφή σχολίων με τα στοιχεία σας (ονοματεπώνυμο, ΑΕΜ, τηλέφωνα και ηλεκτρονικές διευθύνσεις). Επίσης, πριν από κάθε κλάση ή μέθοδο θα υπάρχει επικεφαλίδα σε μορφή σχολίων με σύντομη περιγραφή της λειτουργικότητας του κώδικα. Στην περίπτωση των μεθόδων, πρέπει να περιγράφονται και οι μεταβλητές τους.

Είναι δική σας ευθύνη η απόδειξη καλής λειτουργίας του προγράμματος.

Παραδοτέα για κάθε μέρος της εργασίας

1. Ηλεκτρονική αναφορά που θα περιέχει: εξώφυλλο, περιγραφή του προβλήματος, του αλγορίθμου και των διαδικασιών που υλοποιήσατε και τυχόν ανάλυσή τους.

Σε καμία περίπτωση να μην αντιγράφεται ολόκληρος ο κώδικας μέσα στην αναφορά (εννοείται ότι εξαιρούνται τμήματα κώδικα τα οποία έχουν ως στόχο τη διευκρίνιση του αλγορίθμου)

Προσοχή: Ορθογραφικά και συντακτικά λάθη πληρώνονται.

2. Ένα αρχείο σε μορφή .zip με όνομα “ΑΕΜ1_ΑΕΜ2_PartD.zip”, το οποίο θα περιέχει **όλο** το project σας στο Codeblocks καθώς και το αρχείο της αναφοράς σε pdf (**αυστηρά**). Το αρχείο .zip θα γίνεται upload στο site του μαθήματος **στην ενότητα των ομαδικών εργασιών και μόνο**. Τα ονόματα των αρχείων πρέπει να είναι με **λατινικούς χαρακτήρες**.

Προθεσμία υποβολής

Κώδικας και αναφορά **Κυριακή 3 Ιουνίου, 23:59** (ηλεκτρονικά)

Δε θα υπάρξει καμία παρέκκλιση από την παραπάνω προθεσμία.