**Αντικειμενοστραφής Προγραμματισμός**

**Επεισόδιο ΙΙI**

Κελέση Ελπίδα 9410 [elpidakelesi@ece.auth.gr](mailto:elpidakelesi@ece.auth.gr)

Μυλωνάς Εμμανουήλ 9508 [evmylonas@ece.auth.gr](mailto:evmylonas@ece.auth.gr)

Εικόνα Όταν αυτήν τη φορά ξεκινάς έγκαιρα

Main:

Αρχικά συμπεριλαμβάνουμε τις κατάλληλες κεφαλίδες που είναι απαραίτητες για την υλοποίηση του αρχείου που θέλουμε. Αυτά είναι η Team.h, TeamCompetition.h, ImmunityCompetition.h, CreativeCompetition.h.

Έξω από κάποια συνάρτηση ορίζουμε μια καθολική μεταβλητή τύπου int, την results. Καθολική γιατί θέλουμε να είναι ορατή τόσο στην teamCompetitionDay, ως αποτέλεσμα της συνάρτησης compete, όσο και στην immunityCompetitionDay, όπου μπαίνει ως παράμετρος σε if συνθήκη.

Ξεκινάμε με την normalDay, όπου πολύ απλά με τη χρήση μιας for και των έτοιμων συναρτήσεων που υπάρχουν στην κλάση Ομάδα, πραγματοποιούμε τις δράσεις που ζητά η εκφώνηση και για τις δυο ομάδες.

Ακολουθεί η teamCompetitionDay. Αρχικά, πάλι με βρόγχο οι δύο ομάδες κάνουν κάποιες δράσεις. Έπειτα ζητάμε από το χρήστη να μας δώσει το id και το name του διαγωνισμού αυτού και παράλληλα δημιουργούμε ένα αντικείμενο τύπου FoodAward. Τα τρία αυτά ορίσματα χρησιμοποιούνται στη συνάρτηση αρχικών συνθηκών για τον Ομαδικό Διαγωνισμό που δημιουργούμε μετά. Ακολουθεί η συνάρτηση compete, η οποία, όπως αναφέρθηκε, θα δώσει στη μεταβλητή results είτε την τιμή 1, εάν νίκησε η κόκκινη ομάδα, είτε την τιμή 0, εάν νίκησε η Μπλε ομάδα. Τέλος, πάλι με βρόγχο τελειώνει η μέρα για τις ομάδες.

Έπειτα έχουμε την immunityCompetitionDay. Ίδια δομή με την προηγούμενη συνάρτηση, απλά αντί για FoodAward, έχουμε ImmunityAward και αυτήν τη φορά δημιουργούμε Διαγωνισμό Ασυλίας. Ακολουθεί μια if, η οποία ελέγχει την τιμή της results, ώστε να καλέσει τη συνάρτηση compete με την κατάλληλη ομάδα για όρισμα. Φυσικά, οι ομάδες ακολουθούν την ίδια ρουτίνα πριν τελειώσει η μέρα.

Τέλος, έχουμε την creativeCompetitionDay. Αυτήμ τη φορά ExcurcionAward και (μαντέψτε…) Διαγωνισμός Δημιουργικότητας. Ακολουθεί κλήση της συνάρτησης compete με ορίσματα τις δύο ομάδες και το κλασσικό σβήσιμο της ημέρας.

Competition.h:

Η γενική Κλάση του Διαγωνισμού, τα στοιχεία της οποίας θα κληρονομήσουν αργότερα όλοι οι άλλοι Διαγωνισμοί. Protected μεταβλητές, ώστε να έχουν πρόσβαση και οι άλλες κλάσεις, ο ακέραιος id, και οι string name, winner. Ακολουθούν οι συναρτήσεις αρχικών συνθηκών, μία με ορίσματα και μία χωρίς και η συνάρτηση τελικών συνθηκών. Στη συνάρτηση αρχικών συνθηκών με ορίσματα, δεν υπάρχει θέση για τον νικητή, καθώς είναι άτοπο να ορίζεις τον νικητή στην αρχή. Έπειτα, οι getters και setters για τις τρεις μεταβλητές και τέλος μια status που εκτυπώνει όλες τις τιμές. Αρχείο .cpp δεν χρειάζεται, γιατί όλες οι συναρτήσεις γίνονται inline.

TeamCompetition.h:

Χρησιμοποιούμε την κεφαλίδα Competition.h καθώς θα χρησιμοποιήσουμε στοιχεία από το Διαγωνισμό. Η κλάση του Ομαδικού Διαγωνισμού κληρονομεί τα στοιχεία του Διαγωνισμού και έχει δύο επιπλέον μεταβλητές, μία τύπου FoodAward και μία τύπου Round. Η δεύτερη ορίζεται ως δείκτης, ώστε να δεσμεύσει δυναμικά θέση μνήμης 3 στοιχείων παρακάτω. Στη συνάρτηση αρχικών συνθηκών απλά δημιουργούμε δυναμικά έναν πίνακα 3 θέσεων τύπου Round, ενώ σε αυτή με τα ορίσματα, καλούμε παράλληλα τη συνάρτηση αρχικών συνθηκών του Διαγωνισμού, αναθέτουμε ένα αντικείμενο στο foodAward και δημιουργούμε έναν δυναμικό πίνακα 3 θέσεων για το round. Επειδή ξέρουμε εξαρχής τις παραμέτρους της round (γύρος 1-3, διάρκεια 3 ώρες και κενό στο νικητή, βρισκόμαστε στην αρχή) δημιουργούμε μόνοι μας 3 αντικείμενα με τις παραμέτρους αυτές και τις περνάμε στα στοιχεία που θέλουμε. Ακολουθεί η συνάρτηση τελικών συνθηκών που αποδεσμεύει τον χώρο που ορίστηκε δυναμικά. Στους getters & setters να σημειωθεί πως η getRounds επιστρέφει pointer, στο πλαίσιο του getPlayers, της κλάσης Team. H status χρησιμοποιεί την αντίστοιχη συνάρτηση του Διαγωνισμού και μένει μόνο η compete, που εξηγείται από κάτω.

TeamCompetition.cpp:

To αρχείο περιέχει τον setRounds, και τη συνάρτηση compete. Είναι μια συνάρτηση που καλεί με αναφορά δύο μεταβλητές τύπου Team. Προκειμένου να εξοικειωθούμε με τη δυναμική δέσμευση, όλοι οι πίνακες είναι έτσι ορισμένοι. Έχουμε 2 πίνακες για κάθε ομάδα, ο πρώτος που με την βοήθεια της getPlayers αποθηκεύει τους παίχτες της ομάδας, και ο δεύτερος που θα έχει την τυχαία πεντάδα που θα επιλεγεί. Για να τη δημιουργήσουμε αυτή, χρησιμοποιούμε έναν βρόγχο και τη χρήση της rand. Επίσης απαιτείται έλεγχος, ώστε να μην επιλεγεί ένας παίκτης δύο φορές στην πεντάδα (αναλυτική εξήγηση στον κώδικα). Έπειτα αποθηκεύουμε σε δύο μεταβλητές τη συνολική κούραση και τη συνολική τεχνική της ομάδας και κάνουμε τους ελέγχους που ζητούνται ώστε να κριθεί ο γύρος. Έπειτα, κρίνοντας από το αποτέλεσμα των γυρών με τη χρήση δομής ελέγχου επιστρέφουμε 0 για την ομάδα 2 και 1 για την ομάδα 1. Στην (σπάνια) περίπτωση της ισοπαλίας σε κάποιο γύρο, αλλά και της μοιρασιάς των άλλον δύο, το αποτέλεσμα κρίνεται τυχαία με μια rand που θα επιστρέψει είτε 0 είτε 1.

ImmunityCompetition.h:

Ακριβώς ίδια λογική με τον παραπάνω διαγωνισμό. Κληρονομεί τα χαρακτηριστικά του Διαγωνισμού και έχει μία επιπλέον μεταβλητή τύπου ImmunityAward. Στη συνάρτηση αρχικών συνθηκών χωρίς ορίσματα δεν βάλαμε κάτι στο σκέλος της, γιατί θεωρήσαμε (όπως και στο TeamCompetition) πως παράλληλα καλούνται οι συναρτήσεις αρχικών συνθηκών των Competition και ImmunityAward χωρίς ορίσματα. Μετά την constructor με ορίσματα και τη destructor, ακολουθούν ως γνωστών οι getter & setters, μία status που χρησιμοποιεί τις status των ImmunityAward & Competition και τέλος η πολυαγαπημένη compete.

ImmunityCompetition.cpp:

Υλοποίηση της compete. Όρισμα, κατά αναφορά, μία ομάδα, αυτή που έχασε στον προηγούμενο διαγωνισμό. Δημιουργούμε πάλι δύο δυναμικούς πίνακες. Ο πρώτος περιλαμβάνει όλους τους παίκτες της ομάδας. Με τη χρήση της for αποθηκεύουμε σε κάθε στοιχείο του δεύτερου πίνακα διαδοχικά το αποτέλεσμα της σύνθεσης, με κατάλληλους συντελεστές, τεχνικής και μη-κούρασης (100-κούραση). Ελέγχουμε το αποτέλεσμα αυτό με μια μεταβλητή α που αρχικά είχε την τιμή 0. Εάν το αποτέλεσμα είναι μεγαλύτερο, το α παίρνει την τιμή του αποτελέσματος και το b, την τιμή του συντελεστή του πίνακα εκείνην τη στιγμή. Αφού ο βρόγχος τρέξει όλον τον πρώτο πίνακα, εκτυπώνει το όνομα του παίκτη που βρίσκεται στη θέση b.

Creativity Competition.h :

Η κλάση της Ημέρας Διαγωνισμού Δημιουργικότητας κληρονομεί όλες τις μεταβλητές της κλάσης Competition (id, name, winner), γι αυτό και κάνουμε include το Competition.h . Επίσης περιλαμβάνει τις μεταβλητές, στο protected μέρος της κλασης,  Έπαθλο Εκδρομής, έπαθλο που κερδίζει ο παίκτης που έχει την υψηλότερη τεχνική κατάρτιση, αυξάνει την τεχνική του κατάρτιση κατά 10% και μειώνει τη δημοτικότητά του κατά 10%, και έναν στατικό πίνακα συστατικών, που περιλαμβάνει 10 υλικά μαγειρικής  (αυγά, γάλα, αλεύρι, μακαρόνια, καραβίδα, κουνέλι, βούτυρο, γλυκοπατάτα, ντομάτα, φινόκιο, (είχαμε όρεξη για πρωτοτυπία)) για αν επιλέξει ο παίκτης κάποια από αυτά και να μαγειρέψει.

Στο δημόσιο μέρος της κλάσης περιλαμβάνονται δύο συναρτήσεις αρχικών συνθηκών η μία δεν έχει ορίσματα και η άλλη έχει ως ορίσματα τον κωδικό και το όνομα του αγωνίσματος, αλλά και το Έπαθλο Εκδρομής. Επίσης  περιλαμβάνεται μία συνάρτηση τελικών συνθηκών για να καταστραφούν τα αντικείμενα, καθώς και μία συνάρτηση set και get ExcursionAward ώστε να ορίσουμε και να διαβάσουμε το Excursion Award, μία status για να εκτυπώνει τις τιμές του Competition και το κάθε υλικό του πίνακα των συστατικών. Τέλος περιλαμβάνει τη συνάρτηση compete που έχει ως ορίσματα τις δύο ομάδες και αναπτύσσεται στο CreativityCompetition.cpp (όπου αρχικοποιείται και ο στατικός πίνακας των συστατικών) .

Creativity Competition.cpp :

Όσον αφορά την υλοποίηση της συνάρτησης compete αρχικοποιούμε τη μέγιστη τεχνική (maxTechnique), τον αριθμό των παικτών (playerIndex) και το μέγιστο αριθμό των παικτών (maxIndex). Δημιουργούμε έναν νέο πίνακα και με τους 22 παίκτες, όπου στις πρώτες 11 θέσεις αποθηκεύουμε τον πίνακα της πρώτης ομάδας και στις υπόλοιπες τον πίνακα της δεύτερης. Μετά ελέγχουμε έναν έναν τους παίκτες αυτού του πίνακα ώστε να βρούμε τον παίκτη με τη μέγιστη τεχνική. Εκτυπώνουμε τα στοιχείου του παίκτη με τη μέγιστη τεχνική. Αν ο αριθμός του παίκτη είναι κάτω από 11 τότε ο παίκτης ανήκει στη πρώτη ομάδα, αλλιώς ανήκει στη δεύτερη. Παραλαμβάνει το έπαθλο Εκδρομής και αυξάνει την τεχνική του, αλλά χάνει δημοτικότητα.