

ΕΡΓΑΣΙΑ ΣΤΟΝ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ

ΔΕΥΤΕΡΟ ΠΑΡΑΔΟΤΕΟ

Υλοποίηση βασικών κλάσεων και συναρτήσεων σε C++

Το δεύτερο παραδοτέο της εργασίας απαιτεί τη δημιουργία των βασικότερων κλάσεων του παιχνιδιού του Tichu. Συγκεκριμένα, θα υλοποιηθούν (τμηματικά) οι παρακάτω κλάσεις:

1. Κλάση Card (card.h και card.cpp)

Η κλάση αυτή δημιουργεί τα αντικείμενα των καρτών (φύλλων). Θα πρέπει να περιλαμβάνει μεταβλητές για το είδος (suit) και την τιμή (value) της κάθε κάρτας. Επίσης θα πρέπει να περιλαμβάνει συναρτήσεις get και set για μεταβλητές της κλάσης. Τέλος, θα πρέπει να έχει συναρτήσεις για τον έλεγχο ισότητας με άλλες κάρτες, για το αν η κάρτα μπορεί να συμμετέχει σε συνδυασμούς και για το αν μπορεί να συμμετέχει σε βόμβες.

2. Κλάση Combination (combination.h και combination.cpp)

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών του παιχνιδιού. Θα πρέπει να περιλαμβάνει μεταβλητές για τις κάρτες του συνδυασμού. Επίσης θα πρέπει να περιλαμβάνει συναρτήσεις για την επιστροφή μιας κάρτας του συνδυασμού, για την επιστροφή της τιμής του συνδυασμού (που είναι η τιμή μιας οποιαδήποτε κάρτας του) και τον αριθμό των καρτών του. Τέλος, θα πρέπει να ελέγχεται με συνάρτηση αν ο συνδυασμός μπορεί να παίζεται μετά από κάποιον άλλο συνδυασμό.

3. Κλάση Player (δήλωση στο player.h και υλοποίηση στα player.cpp, playercombinations.cpp)

Η κλάση αυτή δημιουργεί τα αντικείμενα/στιγμιότυπα των παικτών. Θα πρέπει να περιλαμβάνει τις μεταβλητές για το id, το όνομα, την κατάσταση (status) του παίκτη, καθώς επίσης το χέρι (hand) και τη στοίβα με τα φύλλα που έχει μαζέψει. Επίσης θα πρέπει να περιλαμβάνει συναρτήσεις get και set για μεταβλητές της κλάσης. Τέλος, θα πρέπει να περιλαμβάνει συναρτήσεις για τον έλεγχο του αν το χέρι του παίκτη έχει φύλλα, για το αν ο παίκτης έχει το Mah Jong, καθώς και συναρτήσεις για την απόφαση του παίκτη να παίξει και για την επιλογή του ποιο συνδυασμό θα παίξει.

Σημείωση: Κάποιες από τις μεθόδους της κλάσης Player έχουν ήδη υλοποιηθεί στο αρχείο playercombinations.cpp. Εσείς θα πρέπει να υλοποιήσετε τις υπόλοιπες στο αρχείο player.cpp

4. Κλάση Hand (δήλωση στο hand.h και υλοποίηση στα hand.cpp, handcombinations.cpp)

Η κλάση αυτή δημιουργεί τα αντικείμενα/στιγμιότυπα των χεριών των παικτών. Θα πρέπει να περιλαμβάνει τις μεταβλητές για τις κάρτες που έχει στο χέρι του ο παίκτης (και τον αριθμό τους), καθώς επίσης και τους συνδυασμούς που δημιουργεί (και τον αριθμό τους). Περιλαμβάνει επίσης συναρτήσεις για την αρχικοποίηση του χεριού, την προσθήκη/αφαίρεση καρτών, την επιστροφή του πλήθους των καρτών, κ.α. Επίσης, περιλαμβάνονται συναρτήσεις για την επιστροφή των συνδυασμών και του πλήθους τους. Οι συναρτήσεις αυτές είναι υλοποιημένες, συνεπώς η μόνη συνάρτηση που καλείστε να υλοποιήσετε είναι αυτή της κατασκευής των συνδυασμών.

Σημείωση: Όλες σχεδόν οι μέθοδοι της κλάσης Hand έχουν ήδη υλοποιηθεί στο αρχείο hand.cpp. Εσείς θα πρέπει να υλοποιήσετε μόνο τη μέθοδο createAllCombinations στο αρχείο handcombinations.cpp

Όλες οι παραπάνω κλάσεις θα πρέπει να έχουν και συναρτήσεις αρχικών συνθηκών. Κατά τη δημιουργία του παίκτη, το status του θα πρέπει να είναι "HASNTPLAYED" και το μέγεθος της στοίβας του πρέπει να είναι 0. Επίσης, ο παίκτης με id 0 θα πρέπει να λέγεται "Player 1", ενώ ο παίκτης με id 1 θα πρέπει να λέγεται "Player 2", κ.ο.κ. Για την κάρτα το είδος της πρέπει αρχικά να είναι το κενό string και η τιμή της πρέπει αρχικά να είναι ίση με το 0. Τέλος, το πλήθος των καρτών στο combination θα πρέπει να αρχικοποιείται στο 0.

Επιπλέον, θα πρέπει να υλοποιήσετε τις παρακάτω συναρτήσεις στο αρχείο `helpers.cpp`:

- Συνάρτηση `numberOfPlayersThatHaveNoCardsLeft`

Η συνάρτηση αυτή χρησιμοποιείται για να ελεγχθεί αν το παιχνίδι έφτασε στο τέλος του. Παίρνει ως όρισμα έναν πίνακα 4 θέσεων που περιέχει τους παίκτες του παιχνιδιού. Θα πρέπει να υπολογίζει και να επιστρέφει τον αριθμό των παικτών που δεν έχουν καθόλου φύλλα στο χέρι τους.

- Συνάρτηση `lastThreePlayersPassedOrHaveNoCardsLeft`

Η συνάρτηση αυτή χρησιμοποιείται για να ελεγχθεί αν ο γύρος (trick) έφτασε στο τέλος του, ώστε ο παίκτης που είναι η σειρά του να μαζέψει τα φύλλα από το τραπέζι. Παίρνει ως ορίσματα έναν πίνακα 4 θέσεων που περιέχει τους παίκτες του παιχνιδιού και τη σειρά του παίκτη που παίζει. Θα πρέπει να ελέγχει αν οι υπόλοιποι παίκτες έχουν πάει πάσο ή δεν έχουν πλέον φύλλα στο συγκεκριμένο γύρο. Επιστρέφει `true` αν ισχύει η συγκεκριμένη συνθήκη και `false` αν δεν ισχύει.

Σημείωση: Ο παίκτης που είναι η σειρά του να παίξει ορίζεται ως `players[playerTurn]`

Εγκατάσταση

Για να βάλετε το project στο CodeBlocks, αρκεί να αποσυμπίεσετε το αρχείο `Tichu.zip` σε μια τοποθεσία του σκληρού σας δίσκου, και στη συνέχεια από το περιβάλλον του CodeBlocks να επιλέξετε `File → Open...` και να επιλέξετε το `Tichu.cbp` από τον φάκελο που μόλις αποσυμπίεσατε το project.

Παρατηρήσεις

Η υλοποίηση θα πρέπει να γίνει στη C++ και να μπορεί να ανοίξει με το CodeBlocks, **με τις εκδόσεις που χρησιμοποιούμε** στο πλαίσιο του μαθήματος. Ο κώδικάς σας θα πρέπει να είναι καλά τεκμηριωμένος, ώστε να είναι παντού σαφείς οι λεπτομέρειες υλοποίησης.

Για την υλοποίηση, σας δίνονται τα αρχεία κεφαλίδων `.h` των κλάσεων/συναρτήσεων που πρέπει να υλοποιήσετε καθώς και κάποιες βοηθητικές κλάσεις/συναρτήσεις. Επιπλέον, σας δίνεται ο κώδικας της συνάρτησης `main` (`main.cpp`). **Σε καμία περίπτωση δεν επιτρέπεται να επέμβετε στον κώδικα των κλάσεων και των συναρτήσεων αυτών. Σε περίπτωση που το κάνετε, η εργασία σας αυτομάτως θεωρείται λανθασμένη και μηδενίζεται.** Θα πρέπει να γράψετε κώδικα μόνο στα αρχεία `card.cpp`, `combination.cpp`, `player.cpp`, `handcombinations.cpp` και `helpers.cpp`.

Παραδοτέο

Το παραδοτέο θα είναι **ένα αρχείο zip με όνομα `Tichu.zip`** που θα περιλαμβάνει **όλο το project (τα αρχεία που θα υλοποιήσετε αλλά και αυτά που σας έχουν δοθεί)**, δηλαδή ακριβώς ίδιο με το αρχείο `Tichu.zip` που δίνεται, φυσικά **με τον κώδικα υλοποιημένο**. Επιπλέον, προτείνεται πριν δημιουργήσετε το αρχείο zip, να κάνετε `Clean` το project (που γίνεται από το Codeblocks επιλέγοντας στο μενού `Build → Clean`).

Προθεσμία υποβολής

Το παραδοτέο πρέπει να παραδοθεί μέχρι τις **23:59 της Πέμπτης 30 Απριλίου**. Καμία παρέκκλιση δε θα γίνει από την παραπάνω προθεσμία.