

ΕΡΓΑΣΙΑ ΣΤΟΝ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ

ΤΡΙΤΟ ΠΑΡΑΔΟΤΕΟ

Υλοποίηση κληρονομικότητας και συναρτήσεων δεικτών σε C++

Το τρίτο παραδοτέο της εργασίας απαιτεί τη δημιουργία των κληρονομημένων κλάσεων των συνδυασμών του παιχνιδιού του Tichu. Συγκεκριμένα, θα υλοποιηθούν οι παρακάτω κλάσεις **που βρίσκονται όλες στα αρχεία combination.h και combination.cpp**:

1. Κλάση Combination

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών του παιχνιδιού. Θα πρέπει να περιλαμβάνει μεταβλητές για τον τύπο του συνδυασμού, την τιμή (value) του, τον αριθμό των καρτών, καθώς και μια μεταβλητή-διπλό δείκτη που αρχικοποιείται ως πίνακας και περιέχει δείκτες στις κάρτες που περιέχει ο συνδυασμός. Επίσης θα πρέπει να περιλαμβάνει συναρτήσεις για την επιστροφή μιας κάρτας του συνδυασμού, για την επιστροφή της τιμής του συνδυασμού και τον αριθμό των καρτών του, και άλλες βοηθητικές συναρτήσεις.

2. Κλάση Single

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών μονοφυλλίας. Κληρονομεί την κλάση Combination. Θα πρέπει να περιλαμβάνει μια συνάρτηση αρχικών συνθηκών που αρχικοποιεί το συνδυασμό με μία κάρτα.

3. Κλάση Pair

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών ζευγών. Κληρονομεί την κλάση Combination. Θα πρέπει να περιλαμβάνει μια συνάρτηση αρχικών συνθηκών που αρχικοποιεί το συνδυασμό με δύο κάρτες.

4. Κλάση ThreeOfAKind

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών τριών όμοιων καρτών. Κληρονομεί την κλάση Combination. Θα πρέπει να περιλαμβάνει μια συνάρτηση αρχικών συνθηκών που αρχικοποιεί το συνδυασμό με τρεις κάρτες.

5. Κλάση Stairs

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών δύο ή περισσότερων διαδοχικών ζευγών. Κληρονομεί την κλάση Combination. Θα πρέπει να περιλαμβάνει μια συνάρτηση αρχικών συνθηκών που αρχικοποιεί το συνδυασμό με έναν δυναμικό πίνακα από κάρτες (και το πλήθος τους).

6. Κλάση FullHouse

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών φουλ. Κληρονομεί την κλάση Combination. Θα πρέπει να περιλαμβάνει μια συνάρτηση αρχικών συνθηκών που αρχικοποιεί το συνδυασμό με ένα συνδυασμό τριών όμοιων καρτών και ένα συνδυασμό ζεύγους.

7. Κλάση Straight

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών τύπου κέντας. Κληρονομεί την κλάση Combination. Θα πρέπει να περιλαμβάνει μια συνάρτηση αρχικών συνθηκών που αρχικοποιεί το συνδυασμό με έναν δυναμικό πίνακα από κάρτες (και το πλήθος τους). Επίσης, θα πρέπει να έχει μια συνάρτηση που ελέγχει αν όλες οι κάρτες του συνδυασμού έχουν το ίδιο χρώμα (suit).

8. Κλάση FourOfAKind

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών τεσσάρων όμοιων καρτών. Κληρονομεί την κλάση Combination. Θα πρέπει να περιλαμβάνει μια συνάρτηση αρχικών συνθηκών που αρχικοποιεί το συνδυασμό με τέσσερις κάρτες.

9. Κλάση StraightFlush

Η κλάση αυτή δημιουργεί τα αντικείμενα των συνδυασμών τύπου κέντας χρώμα. Κληρονομεί την κλάση Combination. Θα πρέπει να περιλαμβάνει μια συνάρτηση αρχικών συνθηκών που αρχικοποιεί το συνδυασμό με ένα συνδυασμό τύπου κέντας.

Για όλες τις παράγωγες κλάσεις θα πρέπει να ορίσετε σωστά τις συναρτήσεις αρχικών συνθηκών. Μπορείτε να χρησιμοποιήσετε όποια συνάρτηση αρχικών συνθηκών της Combination επιθυμείτε στις συναρτήσεις αρχικών συνθηκών των παράγωγων κλάσεων. Θα πρέπει σε κάθε περίπτωση να αρχικοποιήσετε όλες τις μεταβλητές της βασικής κλάσης Combination, δίνοντας **ιδιαίτερη προσοχή στη μεταβλητή value**, καθώς αυτή θα χρησιμοποιηθεί για να ελέγξετε αν για δύο όμοιους συνδυασμούς ο ένας μπορεί να παιχτεί μετά/πάνω από κάποιον άλλο. Συγκεκριμένα, για τη μεταβλητή value, σε όλες τις περιπτώσεις (Single, Pair, κτλ.) θα πρέπει να είναι **ίση με το μεγαλύτερο φύλλο, εκτός από το FullHouse** όπου θα πρέπει να είναι ίση με το αποτέλεσμα του αθροίσματος Τιμή 3 Όμοιων Φύλλων $\times 10$ + Τιμή 2 Όμοιων Φύλλων. Παράδειγμα: για το FULLHOUSE[5:STAR 5:PAGODA 5:JADE 4:STAR 4:PAGODA] η τιμή της μεταβλητής value θα πρέπει να είναι $5 \times 10 + 4$, δηλαδή ίση με 54.

Επιπλέον, θα πρέπει να υλοποιήσετε τις παρακάτω **συναρτήσεις της κλάσης Player** στο αρχείο player.cpp:

- Συνάρτηση combinationsCanBePlayed

Η συνάρτηση αυτή αναλαμβάνει να ελέγξει αν ένας νέος συνδυασμός (current) μπορεί να παιχτεί πάνω/μετά από τον τελευταίο συνδυασμό (last) που υπάρχει στο τραπέζι. Δέχεται ως είσοδο τον νέο και τον τελευταίο συνδυασμό. Επιστρέφει true **σε οποιαδήποτε από τις παρακάτω περιπτώσεις**:

- Αν ο νέος συνδυασμός είναι ίδιου τύπου με τον τελευταίο συνδυασμό, αν και οι δύο έχουν ίδιο πλήθος καρτών, και αν η τιμή (value) του νέου συνδυασμού είναι μεγαλύτερη από αυτή του τελευταίου συνδυασμού.
- Αν ο νέος συνδυασμός είναι κάποιου τύπου βόμβα, ενώ ο τελευταίος δεν είναι βόμβα.
- Αν ο νέος συνδυασμός είναι βόμβα κέντα-χρώμα (STRAIGHTFLUSH), ενώ ο τελευταίος είναι βόμβα τεσσάρων όμοιων φύλλων (FOUROFAKIND).
- Αν και οι δύο συνδυασμοί είναι βόμβες κέντα-χρώμα (STRAIGHTFLUSH), και ο νέος συνδυασμός έχει περισσότερα φύλλα από τον τελευταίο.

Σε κάθε άλλη περίπτωση επιστρέφει false.

Υπόδειξη: η πρώτη από τις παραπάνω περιπτώσεις θα πρέπει να ισχύει και στην περίπτωση που υπάρχουν συνδυασμοί βόμβες.

- Συνάρτηση countBucketPoints

Η συνάρτηση αυτή μετράει τους πόντους των φύλλων που έχει μαζέψει ο παίκτης στο τέλος του παιχνιδιού. **Οι πόντοι υπολογίζονται με βάση τους παρακάτω κανόνες**:

- Κάθε 5άρι μετράει ως 5 πόντοι.
- Κάθε 10άρι ή Ρήγας μετράει ως 10 πόντοι.
- Ο Δράκος μετράει ως 25 πόντοι.
- Ο Φοίνικας μετράει ως -25 πόντοι (μπορεί να υπάρξει και αρνητικό σκορ).

Όλα τα υπόλοιπα φύλλα δε μετρώνται στους πόντους. Η συνάρτηση θα πρέπει να επιστρέφει το άθροισμα των πόντων για όλα τα φύλλα που έχει μαζέψει ο παίκτης και όχι για τα φύλλα που έχει στο χέρι του.

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 combination.cpp και player.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 Δευτέρας 25 Μαΐου**. Καμία παρέκκλιση δε θα γίνει από την παραπάνω προθεσμία.