

תרגיל 2

Copy Constructor && Assignment

חוובה:

- הגשה בזוגות בלבד
- התוכנית חייבת להתקמפל
- לרשום את השמות שלכם ואת תעודות הזהות בראש כל קובץ בעזרת comments

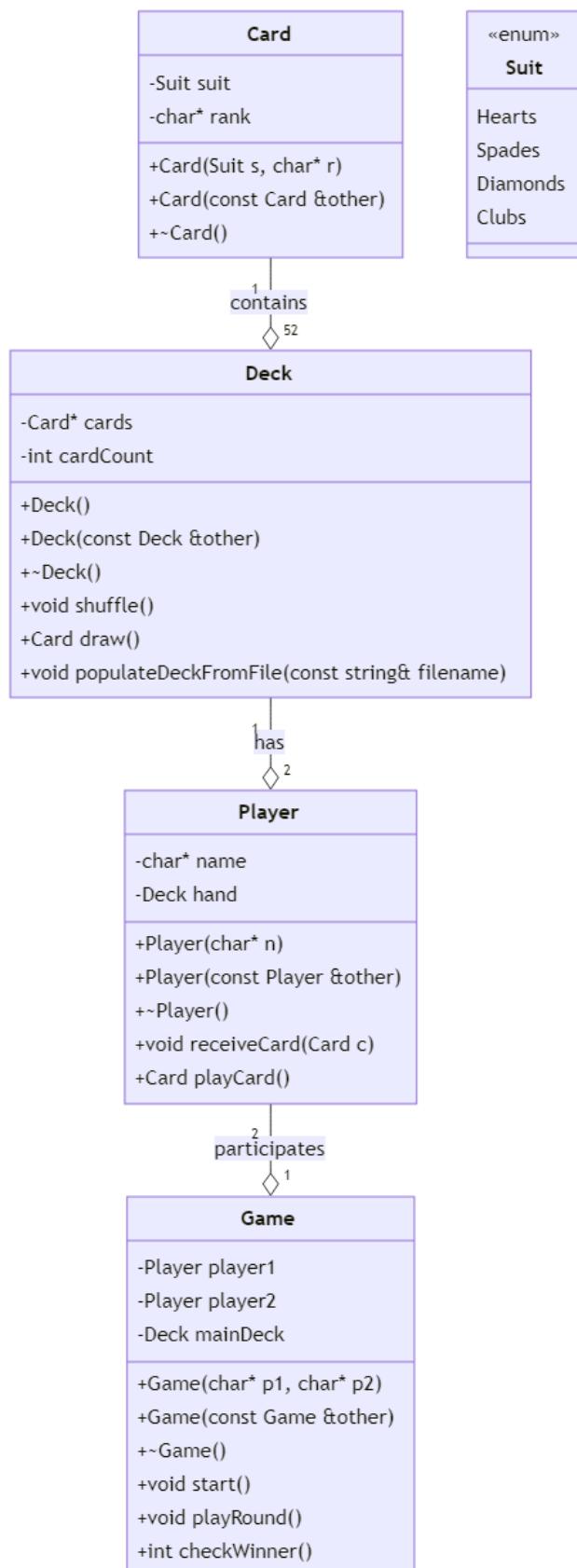
dagshim:

- לא לשכוח לבצע בדיקות ל *link* כנדרש
- לא לשכוח לשחרר זיכרון לאחר שימוש עם הקצת זיכרון או קובץ פתוח
- לחת שמות משמעותיים למשתנים
- הערות (Comments) מעל כל פונקציה המתארות מה היא עשויה

מטרת התרגיל:

בתרגיל זה תתרגלו את שני הנושאים העיקריים של מדרשתם, **Copy Constructor, Assignment Operator**.
ייתה עליכם לבנות מחלקות וABI'קטים על פי ההנחיות מטה, אנא וודאו כי הם עומדים בדרישות!

עליכם ל כתוב את המבנה הבא בקוד:



חשוב: הוספה היא למחרי בידים שלכם ובהתאם Constructors, Destructors, Getters and Setters לשימוש.

הסבר על המחלקות (החלק הזה יהיה באנגלית כי זה מקפץ את השמות בעברית):

Card Class:

- Attributes:
 - Suit suit: Enum value representing the suit of the card. (e.g., hearts, spades).
 - char* rank: Represents the rank of the card (e.g., ace, king).
- Functions:
 - Card(char* s, char* r): Constructor to initialize a card with a suit and rank.
 - Card(const Card &other): Copy constructor for deep copying of card objects.
 - ~Card(): Destructor to free allocated memory.

Deck Class:

- Attributes:
 - Card* cards: Array of Card objects representing the deck.
 - int cardCount: Keeps track of the number of cards in the deck.
- Functions:
 - Deck(): Constructor to initialize the deck with 52 cards.
 - Deck(const Deck &other): Copy constructor for deep copying of deck objects.
 - ~Deck(): Destructor to free allocated memory.
 - void shuffle(): Shuffles the cards in the deck.
 - Card draw(): Draws and returns the top card from the deck.
 - PopulateDeckFromFile(const string& filename): This function is responsible for populating the Deck with Card objects based on the data read from a file.

Player Class:

- Attributes:
 - char* name: Represents the name of the player.
 - Deck hand: Represents the player's hand of cards.
- Functions:
 - Player(char* n): Constructor to initialize a player with a name.
 - Player(const Player &other): Copy constructor for deep copying of player objects.
 - ~Player(): Destructor to free allocated memory.
 - void receiveCard(Card c): Adds a card to the player's hand.
 - Card playCard(): Plays and returns a card from the player's hand.

Game Class:

- Attributes:
 - Player player1, player2: Two players participating in the game.
 - Deck mainDeck: The main deck of cards used in the game.
- Functions:
 - Game(char* p1, char* p2): Constructor to initialize the game with two players.
 - Game(const Game &other): Copy constructor for deep copying of game objects.
 - ~Game(): Destructor to free allocated memory.
 - void start(): Starts the game.
 - void playRound(): Plays a round of the game.
 - int checkWinner(): Checks and returns the winner of the game.

Game

סיפור רקע:

אף פעם לא באמת סימתם משחק קלפים של מלחמה, אך עליו לבקש מהמחשב להראות לכם סימולציות של המשחק



Stole from google

קווים מוחחים:

- חוקים: [https://en.wikipedia.org/wiki/War_\(card_game\)](https://en.wikipedia.org/wiki/War_(card_game))
- גם אם אתם חושבים שאתם יודעים את החוקים, תבדקו שוב
- הבדיקה תעשה על ידי הרצת Main על ידי הבודק
- עליהם לחשב מבחן לקופסא להתמודד עם מקרי קצה שיכולים לקרות שקרו אימם לפונקציות שלכם

דרישות:

- כל מחלוקת בקובץ CPP או CPP משלה
- השליפה של הקלפים חייבת להיות הגונית ועל פי מסויים
- המשחק רץ בעזרת ה Main ואין מגע אדם בזמן ריצה
- שימוש לבשמות הקלפים והכניםו אותם בהתאם, התמודדו עם מקרי קצה

Good luck,
Yuval Ozeri

yuvalozeri@shenkar.ac.il