void GameRun() - входна точка на играта

void GameChoisNewGame(Player\* players) - избор на нова игра или продължаване на съществуваща игра

int ActivePlayersInDealCount(Player\* players) - връща броат на активните играчи в текущата ръка

FileCondition GameReadFromFile(Player\* players) - четене на игра от файл

int GameSetPlayersNum() - задаване на брой играчи при нова игра - потребителски вход

void GameInitPlayers(Player\* players, int playersNum) - първоначално инициализиране на играчите при избор на нова игра

void GameClear(Player\* players) - задава празен списък с играчи

void InitEmptyPlayer(Player& player) - задава празни полета на играч

void ClearDeal(Deal\* deal) - изчиства раздаване. Ако deal.\_dealFlag e true изчиства и подът

GameCondition DealLoop(Player\* players, Deal\* deal) - разиграване на ръка. Вика се рекурсивно до победител в ръка

bool DealStart(Player\* players, Deal\* deal) - раздава картите на играчите и определя състоянието им в зависимост от раздадените карти

void DealPlay(Player\* players, Deal\* deal) - разиграва текущата ръка. Определя състояние на играчите според разиграването им

bool IsPlayerInDeal(player\_condition\_type condition) - определя дали конкретен играч е активен за текущата ръка

FileCondition GameSaveToFile(Player\* players) - записва активните играчи във файл

GameCondition GameLoop(Player\* players) - разиграване на игра. Определя дали играта продължава или се прави запис на състоянието на текущата игра - потребителски вход

int ActivePlayersCount(Player\* players) - връща броят на активните играчи в играта

bool IsPlayerInGame(player\_condition\_type condition) - проверява дали конкретен играч е активен за текущата игра

void DeterminingWinner(Player\* players, Deal\* deal) - определя дали раздаването има победител или има равенство

void SetUpCardDesk(Card\* cardsDeck) – създава ново тесте с карти

std::string CardToString(card\_type card) – връща стринг описващ картата

void SetCards(Player& player, Card\* cardsDesk, int& deckSize) – Раздава на играча карти и задава точките на ръката и стринга изобразяващ ръката

void ActualizePlayers(Player\* players) – локална за GamePlay.cpp. Актуализира списък на активни играчи. Играчите без чипове се сетват с PlayerCondition::Unactive

Локални за Player.cpp

int CalcPoints(Card\* cards) – Изчислява точките на играча за текущата ръка

bool HasSevenClubs(Card\* cards) – проверява за наличие на седмица спатия

int Max(int a, int b) – връща по-голямата от две целочислени променливи

int ThreeSevens(Card\* cards) – връща точките при три седмици

int ThreeOfAKind(Card\* cards) – връща точките при три еднакви карти

int ThreeOfTheSameSuit(Card\* cards) – връща точките при три карти от една боя

int TwoAces(Card\* cards) – връща точките при два асака

int TwoSevens(Card\* cards) – връща точките при две седмици

int TwoCardsOfSuit(Card\* cards) – връща точките при две карти от една боя

int TwoCardsOfPip(Card\* cards) – връща точките при две карти от един и същ вид

int DifferentCards (Card\* cards) – връща точките ако всичките карти са различни, като отчита за наличие на седмица спатия

void GaneratePlayerStringString(Player& player, Card\* cards) – генерира стринговото представяне на картите и точките на играча